Au

BEST AVAILABLE COPY

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-295240

(43)Date of publication of application : 20.10.2000

---

(51)Int.Cl.

H04L 12/28
G06F 13/38
H04L 12/40
H04L 12/56
H04N 7/08
H04N 7/081
H04N 7/24

---

(21)Application number : 11-099457

(22)Date of filing : 06.04.1999

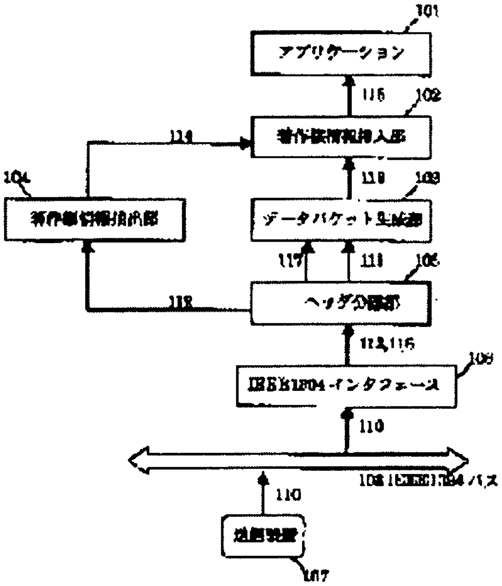
(71)Applicant : MATSUSHITA ELECTRIC IND CO LTD

(72)Inventor : YOSHIDA JIYUNJI
YAMADA MASAZUMI

---

(54) TRANSMITTING DEVICE, RECEIVING DEVICE AND PROGRAM RECORDING MEDIUM

(57)Abstract:



PROBLEM TO BE SOLVED: To simultaneously transmit data and copyright information to an application by which only one kind of data is inputted by separating a transmission packet into a transmission line header and a data block, inserting copyright information extracted from the transmission line header to a data packet which is generated from the data block and constituting an output packet.

SOLUTION: A header separating part 105 separates the data block 111 and an isochronous header 112 from a received isochronous basket 110 and respectively transmits them to a data packet generating part 103 and a copyright information extracting part 104. The part 103 restores and outputs the data packet 113 from the data block 111. The part 104 fetches copyright information 114 from the isochronous header 112 and outputs it. A copyright information inserting part 102 inserts copyright information 114 to a part of a data packet 113 and outputs it to the application 101 as an input/output packet 115.

LEGAL STATUS

[Date of request for examination]	12.03.2002
[Date of sending the examiner's decision of rejection]	
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]	
[Date of final disposal for application]	
[Patent number]	3547641
[Date of registration]	23.04.2004
[Number of appeal against examiner's decision of rejection]	
[Date of requesting appeal against examiner's decision of rejection]	
[Date of extinction of right]	

Copyright (C); 1998,2003 Japan Patent Office

(19)日本国特許庁（ＪＰ）　(12)公開特許公報(A)　(11)特許出願公開番号
特開2000－295240
(P2000－295240A)
(43)公開日　平成12年10月20日(2000.10.20)

(51)Int.Cl.7		識別記号	FI			テーマコード(参考)
H04L	12/28		H04L	11/00	310D	5B077
G06F	13/38	350	G06F	13/38	350	5C059
H04L	12/40		H04L	11/00	320	5C063
	12/56			11/20	102A	5K030
H04N	7/08		H04N	7/08	Z	5K032

審査請求 未請求 請求項の数22 OL (全 14 頁)　最終頁に続く

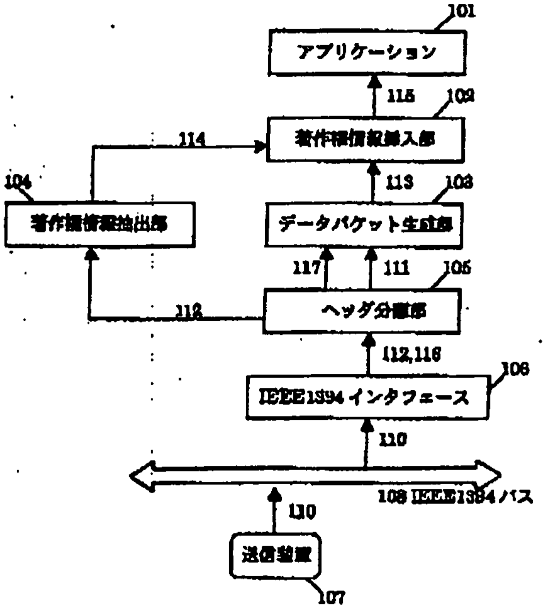
(21)出願番号　特願平11－99457
(22)出願日　平成11年4月6日(1999.4.6)

(71)出願人　000005821
松下電器産業株式会社
大阪府門真市大字門真1006番地
(72)発明者　▲よし▼田 順二
大阪府門真市大字門真1006番地 松下電器産業株式会社内
(72)発明者　山田 正純
大阪府門真市大字門真1006番地 松下電器産業株式会社内
(74)代理人　100092794
弁理士 松田 正道

最終頁に続く

(54)【発明の名称】 送信装置、受信装置及びプログラム記録媒体

(57)【要約】
【課題】1種類のデータのみ入力および/または出力可能なアプリケーションに対し、データと著作権情報の2種類のデータを入力および/または出力することが出来ない。
【解決手段】データパケット113内のシンクバイト602領域に著作権情報114を記述することにより、1種類の入出力パケット115でデータ601と著作権情報114の両方を、アプリケーション101に出力する。


101
アプリケーション
115
102
114
著作権情報挿入部
104
113
103
著作権情報抽出部
データパケット生成部
117
111
105
112
ヘッダ分離部
112,116
106
IEEE1394 インタフェース
110
108 IEEE1394 バス
110
送信装置
107

【特許請求の範囲】

【請求項１】 内部に付加情報が記述されている伝送路ヘッダと、データブロックとを含む伝送パケットを受信するインターフェースと、

前記伝送パケットを前記伝送路ヘッダと前記データブロックとに分離する伝送路ヘッダ分離手段と、

前記伝送路ヘッダから前記付加情報を抽出する付加情報抽出手段と、

前記データブロックから一つまたは複数の前記データブロックを結合させたデータパケットを生成するデータパケット生成手段と、

前記付加情報を前記データパケットに付加または挿入を行い、アプリケーションが同時に処理出来る一種類のデータ形式の出力パケットとして出力する付加情報挿入手段と、を備え、

前記アプリケーションはその付加情報挿入手段で出力された出力パケットを処理することを特徴とする受信装置。

【請求項２】 前記データパケットは、ＭＰＥＧ２データにおける、ソースパケットヘッダとトランスポートパケットとで構成されていることを特徴とする請求項１記載の受信装置。。

【請求項３】 前記付加情報挿入手段は、前記データパケットの先頭もしくは終端に付加情報を付加することを特徴とする請求項１または２記載の受信装置。

【請求項４】 前記付加情報挿入手段は、前記ソースパケットヘッダに前記付加情報を挿入することを特徴とする請求項２記載の受信装置。

【請求項５】 内部に付加情報が記述されている伝送路ヘッダと、データブロックとを含む伝送パケットを受信するインターフェースと、

前記伝送パケットを前記伝送路ヘッダと前記データブロックとに分離する伝送路ヘッダ分離手段と、

前記伝送路ヘッダから前記付加情報を抽出する付加情報抽出手段と、

前記データブロックから一つまたは複数の前記データブロックを結合させたデータパケットを生成するデータパケット生成手段と、

前記データパケットをソースパケットヘッダとソースパケットに分離するソースパケットヘッダ分離手段と、

前記付加情報を前記ソースパケットに付加または挿入または置換を行い、アプリケーションが同時に処理出来る一種類のデータ形式の出力パケットとして出力する付加情報挿入手段と、を備え、

前記アプリケーションは、その付加情報挿入手段で出力された出力パケットを処理することを特徴とする受信装置。

【請求項６】 前記ソースパケットはＭＰＥＧ２データのトランスポートパケットであることを特徴とする請求項５記載の受信装置。

【請求項７】 前記付加情報挿入手段は、前記トランスポートパケットの先頭もしくは終端に前記付加情報を付加することを特徴とする請求項６記載の受信装置。

【請求項８】 前記付加情報挿入手段は、前記トランスポートパケットのシンクバイトを前記付加情報に置き換えることを特徴とする請求項６記載の受信装置。

【請求項９】 前記インタフェースはＩＥＥＥ１３９４であり、前記伝送パケットはアイソクロナスパケットであることを特徴とする請求項１～８のいずれかに記載の受信装置。

【請求項１０】 前記付加情報は、著作権情報であることを特徴とする請求項１～９のいずれかに記載の受信装置。

【請求項１１】 前記付加情報は、アイソクロナスヘッダのＳＹ領域に記述されることを特徴とする請求項１～１０のいずれかに記載の受信装置。

【請求項１２】 アプリケーションが同時に処理出来る一種類の入力パケットを受け取ると、前記入力パケットを構成するデータパケットに追加または挿入されている前記付加情報を取り出す付加情報分離手段と、

前記データパケットから一つまたは複数のデータブロックを生成するデータブロック生成手段と、

前記データブロックに所定の処理を施し、前記付加情報をデータブロックの所定の位置に挿入することにより前記伝送パケットを生成する伝送パケット生成手段と、

前記伝送パケットを送信するインターフェースとを備えたことを特徴とするデータ送信装置。

【請求項１３】 前記データパケットは、ＭＰＥＧ２データにおける、ソースパケットヘッダとトランスポートパケットとで構成されていることを特徴とする請求項１２記載の送信装置。

【請求項１４】 前記入力パケットは、前記データパケットの先頭もしくは終端に前記付加情報を付加されていることを特徴とする請求項１２または１３記載の送信装置。

【請求項１５】 前記入力パケットは、前記ソースパケットヘッダに前記付加情報を挿入することを特徴とする請求項１３の送信装置。

【請求項１６】 アプリケーションが同時に処理出来る一種類のデータ形式の入力パケットを受け取ると、前記入力パケットを構成し、付加または挿入または置換されている付加情報を有するトランスポートパケットから前記付加情報を取り出す付加情報分離手段と、

前記トランスポートパケットにソースパケットヘッダを連結してデータパケットを生成するデータパケット生成手段と、

前記データパケットから一つまたは複数のデータブロックを生成するデータブロック生成手段と、

前記データブロックに所定の処理を施し、前記付加情報をデータブロックの所定の位置に挿入することにより前

記伝送パケットを生成する伝送パケット生成手段と、前記伝送パケットを出力するインターフェースとを備えたことを特徴とする送信装置。

【請求項１７】　前記入力パケットは、前記トランスポートパケットのシンクバイトを付加情報に置き換えたものであることを特徴とする請求項１６記載の送信装置。

【請求項１８】　前記入力パケットは、前記トランスポートパケットの先頭もしくは終端に前記付加情報を付加していることを特徴とする請求項１６記載の送信装置。

【請求項１９】　前記インタフェースはＩＥＥＥ１３９４であり、前記伝送パケットはアイソクロナスパケットであることを特徴とする請求項１２～１８のいずれかに記載の送信装置。

【請求項２０】　前記付加情報は、著作権情報であることを特徴とする請求項１２～１９のいずれかに記載の送信装置。

【請求項２１】　前記付加情報は、アイソクロナスヘッダのＳＹ領域に記述されていることを特徴とする請求項１２～２０のいずれかに記載の送信装置。

【請求項２２】　請求項１～２１のいずれかに記載の受信装置または送信装置を構成する各手段または各構成要素の、一部または全部をコンピュータで実行するためのプログラムを記録したことを特徴とするプログラム記録媒体。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】本発明は著作権情報を含むデータの伝送を行う送信装置および受信装置に関するものである。

【０００２】

【従来の技術】ＬＳＩ技術の向上に伴って映像情報や音声情報をディジタル化して伝送するネットワークが開発されつつある。映像信号や音声信号はリアルタイムで再生される必要があるため、リアルタイム伝送が可能なネットワークが必要となる。

【０００３】このようなリアルタイム伝送に適したネットワークとしてＩＥＥＥ１３９４というネットワークが提案されている。ＩＥＥＥ１３９４はシリアルの高速バスシステムで、データを同期伝送できるため、リアルタイム伝送が可能である。

【０００４】ＩＥＥＥ１３９４は、衛星放送からＭＰＥＧ２データを受信するＳｅｔＴｏｐＢｏｘ（以下ＳＴＢと記述）を始め、多くのディジタル映像音声機器に外部用インタフェースとして搭載することができる。例えばＳＴＢにおいては、ＩＥＥＥ１３９４を用いることにより、外部機器とＳＴＢとの間でのデータ伝送を行うことができる。

【０００５】一方パーソナルコンピュータ（以下ＰＣと記述）においても、標準ＯＳであるＭｉｃｒｏｓｏｆｔ社のＷｉｎｄｏｗｓ９８などに正式にＩＥＥＥ１３９４

がサポートされるようになったことにより、ＰＣの世界でもＩＥＥＥ１３９４は急速に普及しつつある。同時にＭｉｃｒｏｓｏｆｔ社のＤｉｒｅｃｔＳｈｏｗのように、映像音声データなどのストリームデータをＰＣ上で取り扱う環境が整えられている。ＤｉｒｅｃｔＳｈｏｗ上で２種類以上のデータを伝送する場合には、それぞれのデータ伝送は独立に動作するという特徴がある。

【０００６】ところで著作権保護の観点から、ＭＰＥＧ２データなどＡＶデータを伝送する場合には、ＩＥＥＥ１３９４のアイソクロナスヘッダに著作権情報を載せて伝送する方式が、1998年7月に「5C Digital Transmission Content Protection White Paper」において提案されている。この著作権情報を元に例えば受信データをコピーしてよいかどうかを判断している。同時に、正規の受信者以外にAVデータを盗聴されても情報が漏れないようにするために、AVデータの暗号化を行うことがあるが、受信者が暗号を正しく解読するのに必要な情報も著作権情報に含まれている。

【０００７】さて、ＭＰＥＧ２のトランスポートパケット(TSパケット)をIEEE1394上で伝送する方法について図６～１２および図１７を用いて説明する。

【０００８】図６は、TSパケット204の構成である。図６において、601は、映像音声情報などが格納されているデータである。602は、復号器がＴＳパケットの先頭を検出するためのシンクバイトである。

【０００９】図7は、データパケットの構成である。図7において、113はデータパケット、701はソースパケットヘッダである。

【００１０】図8は、ソースパケットヘッダ701の構成である。図8において、801は、各アクセスユニット毎に付けられる復号再生処理用時刻管理タグであるタイムスタンプである。802は予備情報である。

【００１１】図9は、データパケット113の変換例である。図９において、901a,901bはCIPヘッダである。111a,111bはデータブロックである。116a,116bはCIPである。図９では一つのデータパケット１１３が二つのＣＩＰ１１６ａとＣＩＰ１１６ｂに分割されている。一般に伝送速度によって、１個、２個、４個または８個に分割される。

【００１２】図１０は、アイソクロナスパケット110の構成である。図１０において、110は、アイソクロナスパケットである。112は、アイソクロナスヘッダである。116は、CIPである。1001は、ヘッダCRCである。1002は、データCRCである。

【００１３】図１１は、アイソクロナスヘッダ112の構成である。図１１において、1101は、アイソクロナスヘッダの終端に位置するSY領域である。

【００１４】図１２は、著作権情報の構成である。図１２において、114は、著作権情報である。1201はEMI（Ｅｎｃｒｉｐｔｉｏｎ ｍｏｄｅ ｉｎｄｉｃａｔｏｒ）

フラグである。1202は、暗号化されたデータを解読するのに使用される0/Eフラグである。1203は、予備情報である。

【００１５】図１７は、CIPヘッダ901の構成である。図１７において、117は、一つのデータパケット１１３をどのように分割したかを記述する分割情報である。

【００１６】TSパケット204は図６に示されるように、１バイトのシンクバイト602と、187バイトのデータ601とで構成されている。

【００１７】まず、図７に示すように、このTSパケット204にソースパケットヘッダ701を付加し、データパケット113を構成する。ソースパケット701は、図８に示されるように、25ビットのタイムスタンプ801と7ビットの予備情報802で構成されている。タイムスタンプ801にはTSパケット204の伝送タイミングを表す時刻情報が記述されており、予備情報802は将来のために予約された領域で現在は7ビット全てに０が記述されている。

【００１８】次にデータパケット113をいくつかのデータブロックに分割する。分割する個数は、伝送するデータの伝送レートによって決まり、伝送レートが小さいほど分割数は多くなる。このとき分割数は１、２または４である。また伝送レートが高い場合には、逆に複数個のデータパケット113を結合して１個のデータブロックを生成する。

【００１９】図９は、データパケット113の変換例で、１個のデータパケット113を２個のデータブロック111aおよびデータブロック111bに分割している。同時に、データブロック111aにはCIPヘッダ901aを付加し、CIP116aを生成する。このとき、CIPヘッダ901aには、データパケット113を２個のデータブロックに分割したという分割情報117が図１７で示される位置に記録される。

【００２０】最後に図１０に示すように、CIP116に、アイソクロナスヘッダ112とヘッダCRC1001とデータCRC1002を付加し、IEEE1394バス108で伝送形式であるアイソクロナスパケット110を生成する。ヘッダCRC1001は、アイソクロナスヘッダ112の誤り訂正用情報、データCRC1002は、CIP116の誤り訂正用情報である。アイソクロナスヘッダ112は４バイトの大きさを持っており、図１１で示すように、最後の4ビットはSY領域1101と呼ばれる領域になっている。図１２に示すような著作権情報114をデータと一緒に伝送する場合には、SY領域1101に著作権情報114が記述される。EMIフラグ1201はコピー可能かどうかを表す情報であり、0/Eフラグ1202は伝送データが暗号化されている場合に、正常に解読するための情報である。予備情報1203には現在情報は記述されていない。

【００２１】以上説明したＭＰＥＧ２データを受信する受信装置について、図５を用いて説明する。

【００２２】図５は、従来の技術における受信装置の例である。図５において、105は、受け取ったCIP116からデータブロック111と分割情報117を取り出してデータパケット生成部502に出力し、同時に受け取ったアイソクロナスヘッダ112を著作権情報抽出部503に出力するヘッダ分離部である。106は、IEEE1394バス108を通して送信装置107からら送信されてくるアイソクロナスパケット110を受信すると、アイソクロナスパケット110からアイソクロナスヘッダ112とCIP116を取り出して、ヘッダ分離部105に出力する。IEEE1394インタフェース、107は、アイソクロナスパケットやアシンクロナスパケットを送信する送信装置である。108は、IEEE1394バスである。111は、データブロックである。501は、ＤｉｒｅｃｔＳｈｏｗなどのＰＣ上でＭＰＥＧ２データなどを処理するアプリケーションである。502は、受け取った分割情報117を元にデータブロック111からデータパケット113を復元し、アプリケーション501に出力するデータパケット生成部である。503は、受け取ったアイソクロナスヘッダ112から著作権情報114を取り出し、アプリケーション501に出力する著作権情報抽出部である。

【００２３】次にこのような従来の受信装置の動作を説明する。

【００２４】IEEE1394インタフェース106は、IEEE1394バス108を通じて送信装置107から送信されてくるアイソクロナスパケット110を受信すると、アイソクロナスパケット110からアイソクロナスヘッダ112とCIP116を取り出して、ヘッダ分離部105に出力する。

【００２５】ヘッダ分離部は、受け取ったCIP116からデータブロック111と分割情報117を取り出してデータパケット生成部502に出力し、同時に受け取ったアイソクロナスヘッダ112を著作権情報抽出部503に出力する。

【００２６】データパケット生成部502は、受け取った分割情報117を元にデータブロック111からデータパケット113を復元し、アプリケーション501に出力する。

【００２７】著作権情報抽出部503は、受け取ったアイソクロナスヘッダ112から著作権情報114を取り出し、アプリケーション501に出力する。

【００２８】

【発明が解決しようとする課題】しかしながら上記した従来の構成では、アプリケーション501がＤｉｒｅｃｔＳｈｏｗ上のソフトウェアであった場合など、データパケット113がアプリケーション501に出力されるタイミングと、著作権情報114がアプリケーション501に出力されるタイミングが異なる場合には、データパケット113と著作権情報114とを同時に受け取れないという問題点があった。

【００２９】本発明はこのような従来の問題点を鑑みてなされたものであって、著作権情報とデータとを含む伝送パケットを受信し、同時に１種類のデータしか伝送できないアプリケーションに対して、データと著作権情報とを同時に出力できる受信装置を提供することを目的とするものである。

【００３０】また従来の技術では、ＤｉｒｅｃｔＳｈｏ

ｗ上のソフトウェアなどは同時に伝送できるデータの種類は１種類であるため、伝送データと著作権情報という１組のデータをＩＥＥＥ１３９４上に送信する送信装置を実現することはできなかった。

【００３１】本発明はこのような従来の問題点を鑑みてなされたものであって、同時に１種類のデータしか伝送できないアプリケーションから、データと著作権情報とを同時に受取り、著作権情報とデータとを含む伝送パケットとして送信できる送信装置を提供することを目的とするものである。

【００３２】

【課題を解決するための手段】上述した課題を解決するために、第１の本発明（請求項１に対応）は、内部に付加情報が記述されている伝送路ヘッダと、データブロックとを含む伝送パケットを受信するインターフェースと、前記伝送パケットを前記伝送路ヘッダと前記データブロックとに分離する伝送路ヘッダ分離手段と、前記伝送路ヘッダから前記付加情報を抽出する付加情報抽出手段と、前記データブロックから一つまたは複数の前記データブロックを結合させたデータパケットを生成するデータパケット生成手段と、前記付加情報を前記データパケットに付加または挿入を行い、アプリケーションが同時に処理出来る一種類のデータ形式の出力パケットとして出力する付加情報挿入手段と、を備え、前記アプリケーションはその付加情報挿入手段で出力された出力パケットを処理することを特徴とする受信装置である。

【００３３】また、第２の本発明（請求項５に対応）は、内部に付加情報が記述されている伝送路ヘッダと、データブロックとを含む伝送パケットを受信するインターフェースと、前記伝送パケットを前記伝送路ヘッダと前記データブロックとに分離する伝送路ヘッダ分離手段と、前記伝送路ヘッダから前記付加情報を抽出する付加情報抽出手段と、前記データブロックから一つまたは複数の前記データブロックを結合させたデータパケットを生成するデータパケット生成手段と、前記データパケットをソースパケットヘッダとソースパケットに分離するソースパケットヘッダ分離手段と、前記付加情報を前記ソースパケットに付加または挿入または置換を行い、アプリケーションが同時に処理出来る一種類のデータ形式の出力パケットとして出力する付加情報挿入手段と、を備え、前記アプリケーションは、その付加情報挿入手段で出力された出力パケットを処理することを特徴とする受信装置である。

【００３４】また、第３の本発明（請求項１２に対応）は、アプリケーションが同時に処理出来る一種類の入力パケットを受け取ると、前記入力パケットを構成するデータパケットに追加または挿入されている前記付加情報を取り出す付加情報分離手段と、前記データパケットから一つまたは複数のデータブロックを生成するデータブロック生成手段と、前記データブロックに所定の処理を

施し、前記付加情報をデータブロックの所定の位置に挿入することにより前記伝送パケットを生成する伝送パケット生成手段と、前記伝送パケットを送信するインターフェースとを備えたことを特徴とするデータ送信装置である。

【００３５】また、第４の本発明（請求項１６に対応）は、アプリケーションが同時に処理出来る一種類のデータ形式の入力パケットを受け取ると、前記入力パケットを構成し、付加または挿入または置換されている付加情報を有するトランスポートパケットから前記付加情報を取り出す付加情報分離手段と、前記トランスポートパケットにソースパケットヘッダを連結してデータパケットを生成するデータパケット生成手段と、、前記データパケットから一つまたは複数のデータブロックを生成するデータブロック生成手段と、前記データブロックに所定の処理を施し、前記付加情報をデータブロックの所定の位置に挿入することにより前記伝送パケットを生成する伝送パケット生成手段と、前記伝送パケットを出力するインターフェースとを備えたことを特徴とする送信装置である。

【００３６】

【発明の実施の形態】（第１の実施の形態）以下、本発明の第１の実施の形態について、図１、図６～１３および図１７を用いて説明する。従来の技術と同じ番号を付した部分の詳しい説明は省略する。

【００３７】図１は受信装置の例である。図１において、101は、ＤｉｒｅｃｔＳｈｏｗをはじめとする映像音声データなどのストリームデータをＰＣ上で取り扱うことが出来るアプリケーションである。102は、データパケット113の一部に著作権情報114を挿入し、入出力パケット1301を生成する著作権情報挿入部である。103は、受け取った分割情報117を元にデータブロック111からデータパケット113を復元し、著作権情報挿入部102に出力するデータパケット生成部である。104は、受け取ったアイソクロナスヘッダ112のSY領域1101から著作権情報114を取り出し、著作権情報挿入部102に出力する著作権情報抽出部である。115は、入出力パケットである。

【００３８】図１３は、入出力パケットの一例である。図１３において、1301は入出力パケット、1302は予備情報である。

【００３９】次にこのような本実施の形態の動作を説明する。

【００４０】図１において、IEEE1394インタフェース106は、IEEE1394バス108を通して送信装置107から送信されてくるアイソクロナスパケット110を受信すると、アイソクロナスパケット110からアイソクロナスヘッダ112とCIP116を取り出して、ヘッダ分離部105に出力する。

【００４１】ヘッダ分離部は、受け取ったCIP116からデータブロック111と分割情報117を取り出してデータパケ

ット生成部103に出力し、同時に受け取ったアイソクロナスヘッダ112を著作権情報抽出部104に出力する。

【００４２】データパケット生成部103は、受け取った分割情報117を元にデータブロック111からデータパケット113を復元し、著作権情報挿入部102に出力する。

【００４３】著作権情報抽出部104は、受け取ったアイソクロナスヘッダ112のSY領域1101から著作権情報114を取り出し、著作権情報挿入部102に出力する。

【００４４】著作権情報挿入部102は、図１３に示すようにデータパケット113の一部に著作権情報114を挿入し、入出力パケット1301を生成する。すなわち、ソースパケットヘッダ701の予備情報802領域の後ろ４ビットに著作権情報SY114を書き込む。予備情報1302は、将来のための予約領域である。著作権情報挿入部102は、生成した入出力パケット1301をアプリケーション101に出力する。

【００４５】以上のように、本実施の形態においては、ソースパケットヘッダ701の予備情報802に著作権情報114を書き込むことにより、一つの入出力パケット1301で、データと著作権情報の両方を同時にアプリケーション101に出力することが可能となる。

【００４６】なお、本実施の形態の著作権情報114は、予備情報802領域の後ろ４ビットに書き込むとしたが、それ以外の場所に書き込んでも構わない。例えば予備情報８０２領域の先頭４ビットに著作権情報１１４を書き込んでも構わないし、シンクバイト６０２の代わりに著作権情報１１４を書き込んでも構わない。

【００４７】また、本実施の形態の著作権情報114は、SY領域1101以外のアイソクロナスパケットの別の部分に記述されていても構わないし、著作権情報以外の情報を付加されていても構わない。

【００４８】また、本実施の形態の伝送されるデータは、ＭＰＥＧ２のデータとしたが、他のデータであっても構わない。例えばＤＶデータでも同様の処理を行うことが出来る。ＤＶデータを用いる場合、著作権情報１１４は、例えばＤＩＦヘッダの１バイト目にあるＳｅｑ領域（４ビット）に書き込めばよい。

【００４９】また、また本発明のインタフェースはIEEE1394としたが、別のインタフェースであっても構わない。例えば、別のインターフェースの例としては、ＵＳＢを用いることが出来る。

【００５０】また、本発明の受信装置はコンピュータのプログラムによるソフトウェアで構成されていても構わないし、専用の回路を使ったハードウェアで構成されていても構わなし、ソフトウェアとハードウェアの両方を用いて構成されていても構わない。

【００５１】また、本実施の形態のＩＥＥＥ１３９４インタフェースは本発明のインターフェースの例であり、本実施の形態のヘッダ分離部は本発明の伝送路ヘッダ分離手段の例であり、本実施の形態の著作権情報抽出部は

本発明の付加情報抽出手段の例であり、本実施の形態のデータパケット生成部は本発明のデータパケット生成手段の例であり、本実施の形態の著作権情報挿入部は本発明の付加情報挿入手段の例である。

【００５２】（第２の実施の形態）以下、図１の受信装置における入出力パケット115の別の例を、本発明の第２の実施形態として、図１、図６～１２、図１４および図１７を用いて説明する。

【００５３】図１４は、入出力パケットの一例である。図１４において、1401は入出力パケット、1402は予備情報である。

【００５４】次にこのような本実施の形態の動作を説明する。

【００５５】著作権情報挿入部102は、図１４に示すようにデータパケット113の後ろに、著作権情報114と予備情報1402とを付加し、入出力パケット1401を生成する。予備情報1402は、将来のための予約領域である。著作権情報挿入部102は、生成した入出力パケット1401をアプリケーション101に出力する。

【００５６】以上のように、本実施の形態においては、データパケット113と著作権情報114とで一つの入出力パケット1401を構成することで、データと著作権情報の両方を同時にアプリケーション101に出力することが可能となる。

【００５７】なお、本実施の形態の著作権情報114と予備情報1402は、データパケット113の後ろに付加するとしたが、データパケット113の前に付加しても構わないし、著作権情報114と予備情報1402をどのような順序で付加しても構わない。

【００５８】また、本実施の形態の予備情報1402の大きさは４ビットでなくても構わないし、予備情報1402を付加しなくても構わない。

【００５９】また、本実施の形態の著作権情報114は、SY領域1101以外のアイソクロナスパケットの別の部分に記述されていても構わないし、著作権情報以外の情報を付加されていても構わない。

【００６０】また、本実施の形態の伝送されるデータは、ＭＰＥＧ２のデータとしたが、他のデータであっても構わない。例えばＤＶデータでも同様の処理を行うことが出来る。ＤＶデータを用いる場合、著作権情報１１４は、例えばＤＩＦヘッダの１バイト目にあるＳｅｑ領域（４ビット）に書き込めばよい。

【００６１】また、インタフェースはIEEE1394としたが、別のインタフェースであっても構わない。例えば、別のインターフェースとしては、ＵＳＢを用いることが出来る。

【００６２】また、本発明の受信装置は、コンピュータのプログラムによるソフトウェアで構成されていても構わないし、専用の回路を用いたハードウェアで構成されていても構わないし、ハードウェアとソフトウェアの両

方で構成されていても構わない。

【００６３】また、本実施の形態のＩＥＥＥ１３９４インタフェースは本発明のインターフェースの例であり、本実施の形態のヘッダ分離部は本発明の伝送路ヘッダ分離手段の例であり、本実施の形態の著作権情報抽出部は本発明の付加情報抽出手段の例であり、本実施の形態のデータパケット生成部は本発明のデータパケット生成手段の例であり、本実施の形態の著作権情報挿入部は本発明の付加情報挿入手段の例である。

【００６４】（実施の形態３）以下、本発明の第３の実施形態について、図２、図６～１２、図１５および図１７を用いて説明する。従来例および第１の実施形態と同じ番号を付した部分の詳しい説明は省略する。

【００６５】図２は受信装置の例である。図２において、201は、ＤｉｒｅｃｔＳｈｏｗをはじめとする映像音声データなどのストリームデータをＰＣ上で取り扱うことが出来るアプリケーションである。202は、ＴＳパケット２０４の一部に著作権情報114を挿入し、入出力パケット２０５を生成する著作権情報挿入部である。203は、データパケット１１３からソースパケットヘッダ７０１を削除することによってＴＳパケット２０４を生成するTSパケット生成部である。２０５は、アプリケーションに出力される入出力パケットである。

【００６６】図１５は、入出力パケットの一例である。図１５において、1501は入出力パケット、1502は予備情報である。

【００６７】次にこのような本実施の形態の動作を説明する。

【００６８】IEEE1394インタフェース106は、IEEE1394バス108を通して送信装置107から送信されてくるアイソクロナスパケット110を受信すると、アイソクロナスパケット110からアイソクロナスヘッダ112とCIP116を取り出して、ヘッダ分離部105に出力する。

【００６９】ヘッダ分離部は、受け取ったCIP116からデータブロック111と分割情報117を取り出してデータパケット生成部103に出力し、同時に受け取ったアイソクロナスヘッダ112を著作権情報抽出部104に出力する。

【００７０】データパケット生成部103は、受け取った分割情報117を元にデータブロック111からデータパケット113を復元し、TSパケット生成部203に出力する。

【００７１】TSパケット生成部203は、受け取ったデータパケット113から、ソースパケットヘッダ701を削除し、TSパケット204を生成し、著作権情報挿入部202に出力する。

【００７２】著作権情報抽出部104は、受け取ったアイソクロナスヘッダ112のSY領域1101から著作権情報114を取り出し、著作権情報挿入部202に出力する。

【００７３】著作権情報挿入部202は、図１５に示すようにTSパケット204の一部に著作権情報114を挿入し、入出力パケット1501を生成する。すなわち、シンクバイト602領域の後ろ４ビットに著作権情報114を書き込む。予備情報1502は、将来のための予約領域である。著作権情報挿入部202は、生成した入出力パケット1501をアプリケーション201に出力する。

【００７４】以上のように、本実施の形態においては、TSパケット204のシンクバイト602領域に著作権情報114を書き込むことにより、一つの入出力パケット1501で、データと著作権情報の両方を同時にアプリケーション201に出力することが可能となる。

【００７５】なお、本実施の形態の著作権情報114は、シンクバイト602領域の後ろ４ビットに書き込むとしたが、それ以外の場所に書き込んでも構わない。例えばそれ以外の場所としては、シンクバイトの先頭４ビットがある。

【００７６】また、本実施の形態の著作権情報114は、SY領域1101以外のアイソクロナスパケットの別の部分に記述されていても構わないし、著作権情報以外の情報を付加されていても構わない。

【００７７】また、本実施の形態の伝送されるデータは、ＭＰＥＧ２のデータとしたが、他のデータであっても構わない。例えばＤＶデータでも同様の処理を行うことが出来る。ＤＶデータを用いる場合、著作権情報１１４は、例えばＤＩＦヘッダの１バイト目にあるＳｅｑ領域（４ビット）に書き込めばよい。

【００７８】また、本発明のインタフェースはIEEE1394としたが、別のインタフェースであっても構わない。例えば、別のインターフェースの例としては、ＵＳＢを用いることが出来る。

【００７９】また、本発明の受信装置は、コンピュータのプログラムによるソフトウェアで構成されていても構わないし、専用の回路を用いたハードウェアで構成されていても構わないし、ソフトウェアとハードウェアの両方で構成されていても構わない。

【００８０】また、本実施の形態のＩＥＥＥ１３９４インタフェースは本発明のインターフェースの例であり、本実施の形態のヘッダ分離部は本発明の伝送路ヘッダ分離手段の例であり、本実施の形態の著作権情報抽出部は本発明の付加情報抽出手段の例であり、本実施の形態のデータパケット生成部は本発明のデータパケット生成手段の例であり、本実施の形態のＴＳパケット生成部は本発明のソースパケットヘッダ分離手段の例であり、本実施の形態の著作権情報挿入部は本発明の付加情報挿入手段の例である。

【００８１】（第４の実施の形態）以下、図２の受信装置における入出力パケット205の別の例を、本発明の第４の実施形態として、図２、図６～１２および図１６を用いて説明する。

【００８２】図１６は、入出力パケットの一例である。図１６において、1601は入出力パケット、1602は予備情報である。

【００８３】次にこのような本実施の形態の動作を説明する。

【００８４】著作権情報挿入部202は、図１６に示すようにTSパケット204の後ろに、著作権情報114と予備情報1602とを付加し、入出力パケット1601を生成する。予備情報1602は、将来のための予約領域である。著作権情報挿入部202は、生成した入出力パケット1601をアプリケーション201に出力する。

【００８５】以上のように、本実施の形態においては、TSパケット204と著作権情報114とで一つの入出力パケット1601を構成することで、データと著作権情報の両方を同時にアプリケーション201に出力することが可能となる。

【００８６】なお、本実施の形態の著作権情報114と予備情報1602は、TSパケット113の後ろに付加するとしたが、TSパケット204の前に付加しても構わないし、著作権情報114と予備情報1602をどのような順序で付加しても構わない。

【００８７】また、本実施の形態の予備情報1602の大きさは４ビットでなくても構わないし、予備情報1602を付加しなくても構わない。

【００８８】また、本実施の形態の著作権情報１１４は、SY領域1101以外のアイソクロナスパケットの別の部分に記述されていても構わないし、著作権情報以外の情報を付加されていても構わない。

【００８９】また、本実施の形態の伝送されるデータは、ＭＰＥＧ２のデータとしたが、他のデータであっても構わない。例えばＤＶデータでも同様の処理を行うことが出来る。ＤＶデータを用いる場合、著作権情報１１４は、例えばＤＩＦヘッダの１バイト目にあるＳｅｑ領域（４ビット）に書き込めばよい。

【００９０】また、本発明のインタフェースはIEEE1394としたが、別のインタフェースであっても構わない。例えば別のインターフェースの例としては、ＵＳＢを用いることが出来る。

【００９１】また、本発明の受信装置は、コンピュータのプログラムによるソフトウェアで構成されていても構わないし、専用の回路を用いたハードウェアで構成されていても構わないし、ソフトウェア及びハードウェアを用いて構成されていても構わない。

【００９２】また、本実施の形態のＩＥＥＥ１３９４インタフェースは本発明のインターフェースの例であり、本実施の形態のヘッダ分離部は本発明の伝送路ヘッダ分離手段の例であり、本実施の形態の著作権情報抽出部は本発明の付加情報抽出手段の例であり、本実施の形態のデータパケット生成部は本発明のデータパケット生成手段の例であり、本実施の形態のＴＳパケット生成部は本発明のソースパケットヘッダ分離手段の例であり、本実施の形態の著作権情報挿入部は本発明の付加情報挿入手段の例である。

【００９３】（第５の実施の形態）以下、本発明の第５の実施形態について、図３および図６〜１３を用いて説明する。従来例と同じ番号を付した部分の詳しい説明は省略する。

【００９４】図３は本実施の形態の送信装置の例である。図３において、301は、ＤｉｒｅｃｔＳｈｏｗをはじめとする映像音声データなどのストリームデータをＰＣ上で取り扱うことが出来るアプリケーションである、302は、入出力パケット１１５から著作権情報１１４を抽出し、データパケット１１３を出力する著作権情報抽出部である。303は、データパケット１１３を分割し、分割情報とデータブロックを出力するデータブロック生成部である。304は、分割情報１１７とデータブロックからＣＩＰを生成するCIP生成部である。305はIEEE1394インタフェースである。306は、受信装置である。

【００９５】次にこのような本実施の形態の動作を説明する。

【００９６】アプリケーション301は、図１３に示すように、ソースパケットヘッダ701領域に著作権情報114と予備情報1301とが記述されている入出力パケット1301を、入出力パケット115として著作権情報抽出部302に出力する。

【００９７】著作権情報抽出部302は、受け取った入出力パケット1301から著作権情報114を取り出し、CIP生成部304に出力する。同時に入出力パケット1301に記録されているタイムスタンプ801からソースパケットヘッダ701を作成し、TSパケット204と作成したソースパケットヘッダとでデータパケット113を作成し、データブロック生成部303に出力する。

【００９８】データブロック生成部303は、受け取ったデータパケット113を、図９のように、１個もしくは複数個のデータブロック111に分割する。もしくは、複数個のデータパケット113を連結して、１個のデータブロック111を生成する。データブロック生成部303は、生成したデータブロック111と、データパケット113の分割情報とを、CIP生成部304に出力する。

【００９９】CIP生成部304は、受け取った分割情報117を元に図１７に示すようなCIPヘッダ901を生成し、データブロック111の先頭に生成したCIPヘッダ901を付加したものをCIP116としてIEEE1394インタフェース305に出力する。このとき同時に著作権情報抽出部302から受け取った著作権情報114を、SY領域1101としてIEEE1394インタフェース305に出力する。

【０１００】IEEE1394インタフェース305は、受け取った著作権情報114を元に図１１のようなアイソクロナスヘッダ112を生成し、またアイソクロナスヘッダ112からヘッダCRC1001を生成する。同時に受け取ったCIP116からデータCRC1002を生成し、図１０に示すようにアイソクロナスヘッダ112とヘッダCRC1001とCIP116とデータCRC1002とからアイソクロナスパケット110を生成し、IEEE

1394バス108に出力する。

【０１０１】以上のように、本実施の形態においては、ソースパケットヘッダ701の予備情報802領域に著作権情報114を書き込んだ入出力パケット1301を、送信装置への入力とすることで、アプリケーション301はデータと著作権情報の両方を同時に受信装置に伝送することが可能となる。

【０１０２】なお、本実施の形態の著作権情報114は、予備情報802領域の後ろ４ビットに書き込むとしたが、それ以外の場所に書き込んでも構わない。例えば予備情報８０２領域の先頭４ビットに著作権情報１１４を書き込んでも構わないし、シンクバイト６０２の代わりに著作権情報１１４を書き込んでも構わない。

【０１０３】また、本実施の形態の著作権情報はアイソクロナスパケットの別の部分に記述されていても構わないし、著作権情報以外の情報を付加されていても構わない。

【０１０４】また、データブロック生成部303では、１個のデータパケット113から１個もしくは複数のデータブロックに分割しても構わないし、２個以上のデータパケット113を連結して１個のデータブロックを構成しても構わない。

【０１０５】また、伝送されるデータは、ＭＰＥＧ２のデータとしたが、他のデータであっても構わない。例えばＤＶデータでも同様の処理を行うことが出来る。ＤＶデータを用いる場合、著作権情報１１４は、例えばＤＩＦヘッダの１バイト目にあるＳｅｑ領域（４ビット）に書き込めばよい。

【０１０６】また、本発明のインタフェースはIEEE1394としたが、別のインタフェースであっても構わない。例えば別のインターフェースの例としては、ＵＳＢを用いることが出来る。

【０１０７】また、本発明の送信装置は、コンピュータのプログラムによるソフトウェア構成されていてもかまわないし、専用の回路を用いたハードウェアで構成されておいても構わないし、ソフトウェア及びハードウェアの両方で構成されていても構わない。

【０１０８】また本実施の形態の著作権情報抽出部は本発明の付加情報分離手段の例であり、本実施の形態のデータブロック生成部は本発明のデータブロック生成手段の例であり、本実施の形態のＣＩＰ生成部は本発明の伝送パケット生成手段の例であり、本実施の形態のＩＥＥＥ１３９４インタフェースは本発明のインターフェースの例である。

【０１０９】（第６の実施の形態）以下、図３の送信装置における入出力パケット115の別の例を、本発明の第６の実施の形態として、図３、図６～１２、図１４および図１７を用いて説明する。

【０１１０】次にこのような本実施の形態の動作を説明する。

【０１１１】アプリケーション301は、図１４に示すように、データパケット113の後ろに著作権情報114と予備情報1402とを付加されている入出力パケット1401を、入出力パケット115として著作権情報抽出部302に出力する。

【０１１２】著作権情報抽出部302は、受け取った入出力パケット1401から著作権情報114を取り出し、CIP生成部304に出力する。同時に入出力パケット1401からデータパケット113を取り出し、データブロック生成部303に出力する。

【０１１３】以下、第５の実施の形態と同じ動作を行う。

【０１１４】以上のように、本実施の形態においては、データパケット113の後ろに著作権情報114と予備情報1402とを付加されている入出力パケット1401を、送信装置への入力とすることで、アプリケーション301はデータと著作権情報の両方を同時に受信装置に伝送することが可能となる。

【０１１５】なお、本実施の形態の著作権情報114と予備情報1402は、データパケット113の後ろに付加するとしたが、データパケット113の前に付加しても構わないし、著作権情報114と予備情報1402をどのような順序で付加しても構わない。

【０１１６】また、本実施の形態の予備情報1402の大きさは４ビットでなくても構わないし、予備情報1402を付加しなくても構わない。

【０１１７】また、本実施の形態の著作権情報はアイソクロナスパケットの別の部分に記述されていても構わないし、著作権情報以外の情報を付加されていても構わない。

【０１１８】また、本実施の形態のデータブロック生成部303では、１個のデータパケット113から１個もしくは複数のデータブロックに分割しても構わないし、２個以上のデータパケット113を連結して１個のデータブロックを構成しても構わない。

【０１１９】また、本実施の形態の伝送されるデータは、ＭＰＥＧ２のデータとしたが、他のデータであっても構わない。例えばＤＶデータでも同様の処理を行うことが出来る。ＤＶデータを用いる場合、著作権情報１１４は、例えばＤＩＦヘッダの１バイト目にあるＳｅｑ領域（４ビット）に書き込めばよい。

【０１２０】また、本発明のインタフェースはIEEE1394としたが、別のインタフェースであっても構わない。例えば別のインターフェースの例としては、ＵＳＢを用いることが出来る。

【０１２１】また、本発明の送信装置は、コンピュータのプログラムによるソフトウェアで構成されていても構わないし、専用の回路を用いたハードウェアで構成されておいても構わないし、ソフトウェア及びハードウェアの両方で構成されていても構わない。

【０１２２】また、本発明の送信装置は、コンピュータのプログラムによるソフトウェア構成されていてもかまわないし、専用の回路を用いたハードウェアで構成されておいても構わないし、ソフトウェア及びハードウェアの両方で構成されていても構わない。

【０１２３】また本実施の形態の著作権情報抽出部は本発明の付加情報分離手段の例であり、本実施の形態のデータブロック生成部は本発明のデータブロック生成手段の例であり、本実施の形態のＣＩＰ生成部は本発明の伝送パケット生成手段の例であり、本実施の形態のＩＥＥＥ１３９４インタフェースは本発明のインターフェースの例である。

【０１２４】（第７の実施の形態）以下、本発明の第７の実施形態について、図４、図６～１２および図１５を用いて説明する。従来例および第５の実施形態と同じ番号を付した部分の詳しい説明は省略する。

【０１２５】図４は本実施の形態の送信装置の例である。

【０１２６】図４において、401は、ＤｉｒｅｃｔＳｈｏｗをはじめとする映像音声データなどのストリームデータをＰＣ上で取り扱うことが出来るアプリケーションである。402は入出力パケット２０５から著作権情報１１４を抽出し、ＴＳパケット２０４を出力する著作権情報抽出部である。403はタイムスタンプ生成部である。

【０１２７】次にこのような本実施の形態の動作を説明する。

【０１２８】アプリケーション401は、図１５に示すように、TSパケット204のシンクバイト602領域に著作権情報114と予備情報1502を記述されている入出力パケット1501を、入出力パケット205として著作権情報抽出部402に出力する。

【０１２９】著作権情報抽出部402は、受け取った入出力パケット1501から著作権情報114を取り出し、CIP生成部304に出力する。同時に入出力パケット1501に記録されている著作権情報114領域および予備情報1502領域に、シンクバイト602を書き込み、TSパケット204を生成しタイムスタンプ生成部403に出力する。

【０１３０】タイムスタンプ生成部403は、タイムスタンプ生成部403にTSパケット204が到着した時刻からタイムスタンプ801を作成し、図７および図８に示すようにタイムスタンプ801と予備情報802とを組み合わせたソースパケットヘッダ701をTSパケット204に付加し、データパケット113としてデータブロック生成部303に出力する。

【０１３１】以下、第５の実施の形態と同じ動作を行う。

【０１３２】以上のように、本実施の形態においては、TSパケット204のシンクバイト602領域に著作権情報114と予備情報1502を記述されている入出力パケット1501を、送信装置への入力とすることで、アプリケーション401はデータと著作権情報の両方を同時に受信装置に伝送することが可能となる。

【０１３３】なお、本実施の形態の著作権情報114は、予備情報802領域の後ろ４ビットに書き込むとしたが、それ以外の場所に書き込んでも構わない。例えば予備情報８０２領域の先頭４ビットに著作権情報１１４を書き込んでも構わないし、シンクバイト６０２の代わりに著作権情報１１４を書き込んでも構わない。

【０１３４】また、本実施の形態の著作権情報はアイソクロナスパケットの別の部分に記述されていても構わないし、著作権情報以外の情報を付加されていても構わない。

【０１３５】また、本実施の形態のデータブロック生成部303では、１個のデータパケット113から１個もしくは複数のデータブロックに分割しても構わないし、２個以上のデータパケット113を連結して１個のデータブロックを構成しても構わない。

【０１３６】また、本実施の形態の伝送されるデータは、ＭＰＥＧ２のデータとしたが、他のデータであっても構わない。例えばＤＶデータでも同様の処理を行うことが出来る。ＤＶデータを用いる場合、著作権情報１１４は、例えばＤＩＦヘッダの１バイト目にあるＳｅｑ領域（４ビット）に書き込めばよい。

【０１３７】また、本発明のインタフェースはIEEE1394としたが、別のインタフェースであっても構わない。例えば別のインターフェースの例としては、ＵＳＢを用いることが出来る。

【０１３８】また、本発明の送信装置は、コンピュータのプログラムによるソフトウェアで構成されていても構わないし、専用の回路を用いたハードウェアで構成されていても構わないし、ソフトウェア及びハードウェアの両方で構成されていても構わない。

【０１３９】また、本実施の形態の著作権情報抽出部は本発明の付加情報分離手段の例であり、本実施の形態のタイムスタンプ生成部は本発明のデータパケット生成手段の例であり、本実施の形態のデータブロック生成部は本発明のデータブロック生成手段の例であり、本実施の形態のＣＩＰ生成部は本発明の伝送パケット生成手段の例であり、本実施の形態のＩＥＥＥ１３９４インタフェースは本発明のインターフェースの例である。

【０１４０】（第８の実施の形態）以下、図４の送信装置における入出力パケット205の別の例を、本発明の第８の実施形態として、図４、図６～１２、図１６および図１７を用いて説明する。

【０１４１】以下にこのような本実施の形態の動作を説明する。

【０１４２】アプリケーション301は、図１６に示すように、TSパケット204の後ろに著作権情報114と予備情報1602とを付加されている入出力パケット1601を、入出力パケット205として著作権情報抽出部302に出力する。

【０１４３】著作権情報抽出部402は、受け取った入出力パケット1501から著作権情報114を取り出し、CIP生成部304に出力する。同時に入出力パケット1501からTSパケット204を取り出し、タイムスタンプ生成部403に出力する。

【０１４４】以下、第７の実施の形態と同じ動作を行う。

【０１４５】以上のように、本実施の形態においては、TSパケット204の後ろに著作権情報114と予備情報1602とを付加されている入出力パケット1601を、送信装置への入力とすることで、アプリケーション401はデータと著作権情報の両方を同時に送信装置に伝送することが可能となる。

【０１４６】なお、本実施の形態の著作権情報114と予備情報1602は、TSパケット204の後ろに付加するとしたが、TSパケット204の前に付加しても構わないし、著作権情報114と予備情報1602をどのような順序で付加しても構わない。

【０１４７】また、本実施の形態の予備情報1602の大きさは４ビットでなくても構わないし、予備情報1602を付加しなくても構わない。

【０１４８】また、本実施の形態の著作権情報はアイソクロナスパケットの別の部分に記述されていても構わないし、著作権情報以外の情報を付加されていても構わない。

【０１４９】また、本実施の形態のデータブロック生成部303では、１個のデータパケット113から１個もしくは複数のデータブロックに分割しても構わないし、２個以上のデータパケット113を連結して１個のデータブロックを構成しても構わない。

【０１５０】また、本実施の形態の伝送されるデータは、ＭＰＥＧ２のデータとしたが、他のデータであっても構わない。例えばＤＶデータでも同様の処理を行うことが出来る。ＤＶデータを用いる場合、著作権情報１１４は、例えばＤＩＦヘッダの１バイト目にあるＳｅｑ領域（４ビット）に書き込めばよい。

【０１５１】また、また本発明のインタフェースはIEEE1394としたが、別のインタフェースであっても構わない。例えば別のインターフェースの例としては、ＵＳＢを用いることが出来る。

【０１５２】また、本発明の送信装置は、コンピュータのプログラムによるソフトウェアで構成されていても構わないし、専用の回路を用いたハードウェアで構成されていても構わないし、ソフトウェア及びハードウェアの両方で構成されていても構わない。

【０１５３】また、本実施の形態の著作権情報抽出部は本発明の付加情報分離手段の例であり、本実施の形態のタイムスタンプ生成部は本発明のデータパケット生成手段の例であり、本実施の形態のデータブロック生成部は本発明のデータブロック生成手段の例であり、本実施の形態のＣＩＰ生成部は本発明の伝送パケット生成手段の例であり、本実施の形態のＩＥＥＥ１３９４インタフェースは本発明のインターフェースの例である。

【０１５４】また、本発明の受信装置または送信装置を構成する各手段または各構成要素の、全部または一部の機能をコンピュータに実行させるためのプログラムを格納したプログラム記録媒体も本発明に属する。

【０１５５】

【発明の効果】以上説明したところから明らかなように、本発明は、同時に１種類のデータしか入力できないアプリケーションに対して、データと著作権情報の両方を同時に伝送することが出来る受信装置を提供することが出来る。

【０１５６】また、本発明は、同時に１種類のデータしか出力できないアプリケーションに対して、データと著作権情報の両方を同時に伝送することが出来る送信装置を提供することが出来る。

【図面の簡単な説明】

【図１】本発明の第１の実施形態および第２の実施形態における受信装置の構成を示すブロック図

【図２】本発明の第３の実施形態および第４実施形態における受信装置の構成を示すブロック図

【図３】本発明の第５の実施形態および第６の実施形態における送信装置の構成を示すブロック図

【図４】本発明の第７の実施形態および第８の実施形態における送信装置の構成を示すブロック図

【図５】従来の技術における受信装置の構成を示すブロック図

【図６】TSパケット204の構成を示す図

【図７】データパケットの構成を示す図

【図８】ソースパケットヘッダ701の構成を示す図

【図９】データパケット113の変換例を示す図

【図１０】アイソクロナスパケット110の構成を示す図

【図１１】アイソクロナスヘッダ112の構成を示す図

【図１２】著作権情報の構成を示す図

【図１３】本発明の第１の実施形態および第５の実施形態における入出力パケット115の例を示す図

【図１４】本発明の第２の実施形態および第６の実施形態における入出力パケット115の例を示す図

【図１５】本発明の第３の実施形態および第７の実施形態における入出力パケット205の例を示す図

【図１６】本発明の第４の実施形態および第８の実施形態における入出力パケット205の例を示す図

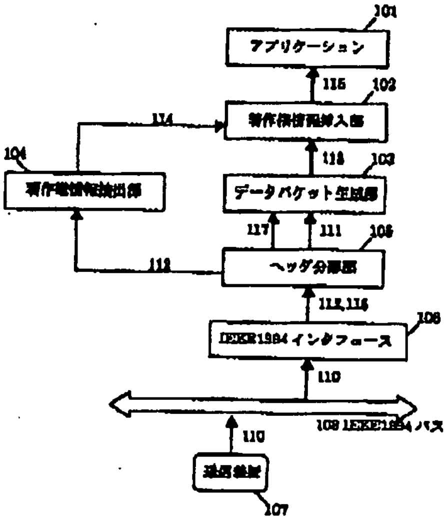
【図１７】CIPヘッダ901の構成を示す図

【符号の説明】

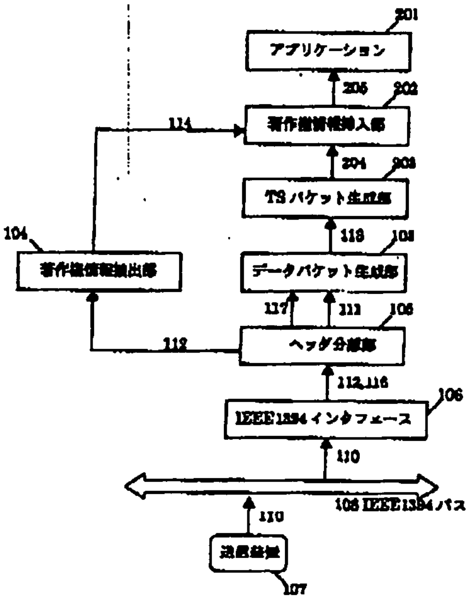
101	アプリケーション
102	著作権情報挿入部
103	データパケット生成部
104	著作権情報抽出部
105	ヘッダ分離部

106	IEEE1394インタフェース
107	送信装置
108	IEEE1394バス
110	アイソクロナスパケット
111, 111a, 111b	データブロック
112	アイソクロナスヘッダ
113	データパケット
114	著作権情報
115	入出力パケット
116, 116a, 116b	CIP
117	分割情報
201	アプリケーション
202	著作権情報挿入部
203	TSパケット生成部
204	TSパケット
205	入出力パケット
301	アプリケーション
302	著作権情報抽出部
303	データブロック生成部
304	CIP生成部
305	IEEE1394インタフェース
306	受信装置
401	アプリケーション
402	著作権情報抽出部
403	タイムスタンプ生成部
501	アプリケーション
601	データ
602	シンクバイト
701	ソースパケットヘッダ
801	タイムスタンプ
802	予備情報
901a, 901b	CIPヘッダ
1001	ヘッダCRC
1002	データCRC
1101	SY領域
1201	EMIフラグ
1202	O/Eフラグ
1203	予備情報
1301	入出力パケット
1302	予備情報
1401	入出力パケット
1402	予備情報
1501	入出力パケット
1502	予備情報
1601	入出力パケット
1602	予備情報

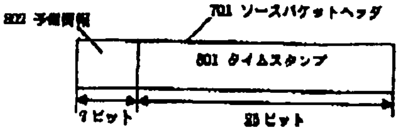
【図１】


101
アプリケーション
115
102
著作権情報挿入部
114
104
著作権情報抽出部
113
103
データパケット生成部
117
111
105
112
ヘッダ分離部
112,116
106
IEEE1394インタフェース
110
108 IEEE1394バス
110
送信装置
107

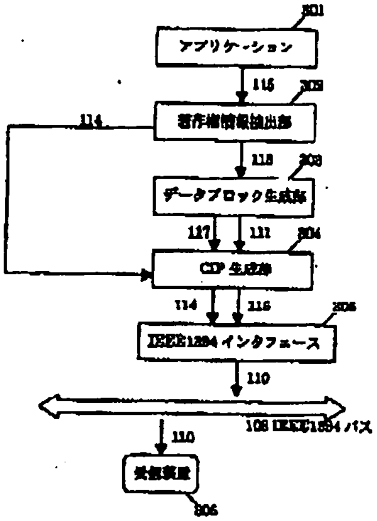
【図２】


201
アプリケーション
205
202
著作権情報挿入部
114
204
203
TSパケット生成部
113
103
104
著作権情報抽出部
データパケット生成部
117
111
105
112
ヘッダ分離部
112,116
106
IEEE1394インタフェース
110
108 IEEE1394バス
110
送信装置
107

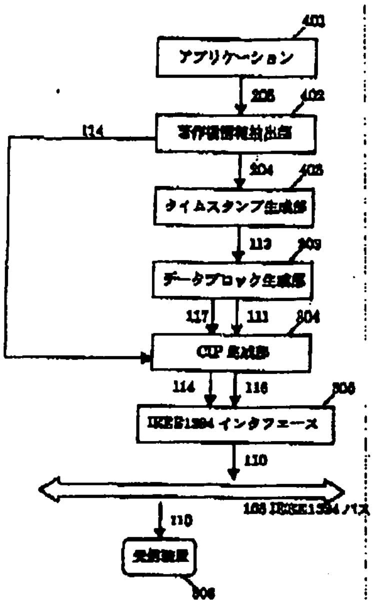
【図８】


802 予備情報
701 ソースパケットヘッダ
801 タイムスタンプ
7ビット
25ビット

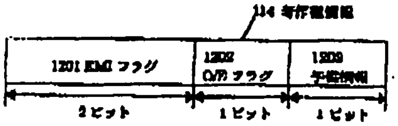
【図３】


301
アプリケーション
115
302
著作権情報抽出部
114
116
303
データブロック生成部
117
111
304
CIP生成部
114
116
305
IEEE1394インタフェース
110
108 IEEE1394バス
110
受信装置
306

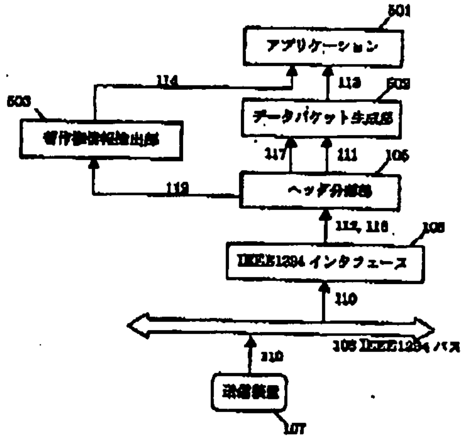
【図４】


401
アプリケーション
205
402
著作権情報抽出部
114
204
403
タイムスタンプ生成部
113
303
データブロック生成部
117
111
304
CIP生成部
114
116
305
IEEE1394インタフェース
110
108 IEEE1394バス
110
受信装置
306

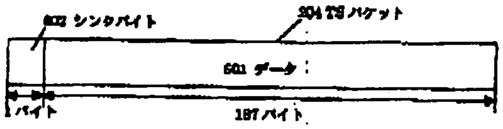
【図１２】


114 著作権情報
1201 EMIフラグ
1202 O/Eフラグ
1203 予備情報
2ビット
1ビット
1ビット

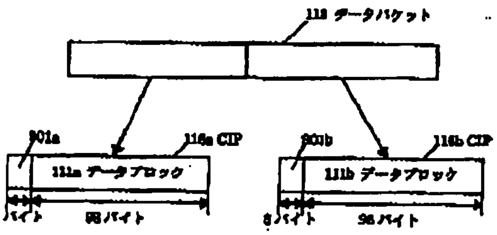
【図５】


501
アプリケーション
114
113
502
503
著作権情報抽出部
データパケット生成部
117
111
106
119
ヘッダ分離部
112, 116
105
IEEE1394インタフェース
110
108 IEEE1394バス
110
送信装置
107

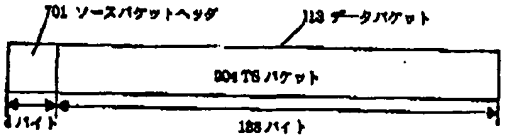
【図６】


602 シンクバイト
204 TSパケット
601 データ
1バイト
187バイト

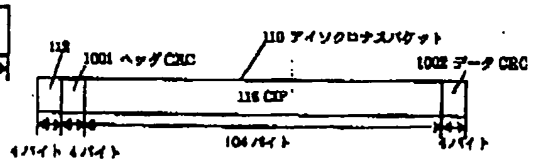
【図９】


113 データパケット
901a
116a CIP
111a データブロック
8バイト
96バイト
901b
116b CIP
111b データブロック
8バイト
96バイト

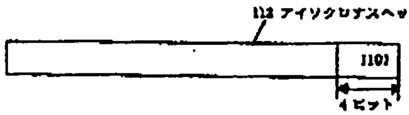
【図７】


701 ソースパケットヘッダ
113 データパケット
204 TSパケット
4バイト
188バイト

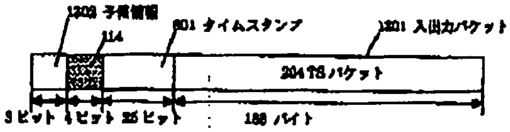
【図１０】


112
1001 ヘッダCRC
110 アイソクロナスパケット
1002 データCRC
116 CIP
4バイト
4バイト
104バイト
4バイト

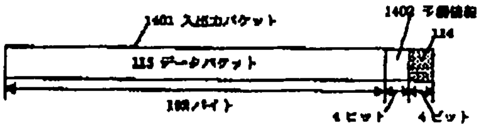
【図１１】


112 アイソクロナスヘッダ
1101
4ビット

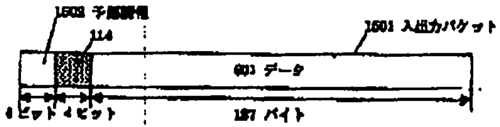
【図１３】


1302 予備情報
114
801 タイムスタンプ
1301 入出力パケット
204 TS パケット
3ビット
4ビット
25ビット
188 バイト

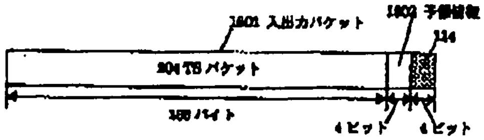
【図１４】


1401 入出力パケット
1402 予備情報
114
115 データパケット
188バイト
4ビット
4ビット

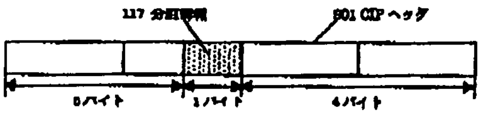
【図１５】


1502 予備情報
114
1501 入出力パケット
803 データ
4ビット
4ビット
187 バイト

【図１６】


1601 入出力パケット
1602 予備情報
114
204 TS パケット
188 バイト
4ビット
4ビット

【図１７】


117 分割情報
801 CIP ヘッダ
3バイト
1バイト
4バイト

フロントページの続き

(51) Int. Cl.⁷	識別記号	F I	テーマコード(参考)
Ｈ０４Ｎ　7/081		Ｈ０４Ｎ　7/13	Ｚ　５Ｋ０３３
7/24			

Ｆターム(参考)　5B077 FF01 FF13 NN02
5C059 KK43 MA00 RB02 RB09 RC35 UA29
5C063 AB03 AB07 AC10 CA23 CA34 CA36
5K030 GA15 HB00 HB19 JA05 KA02 LA08 LD07
5K032 BA16 CD01 DB19 DB22
5K033 BA15 CC01 DA13 DB12 DB14

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 172 723 A1

(12) **EUROPEAN PATENT APPLICATION**
published in accordance with Art. 158(3) EPC

(43) Date of publication:
16.01.2002 Bulletin 2002/03

(21) Application number: 00912894.3

(22) Date of filing: 29.03.2000

(51) Int Cl.7: G06F 3/06

(86) International application number:
PCT/JP00/01926

(87) International publication number:
WO 00/60445 (12.10.2000 Gazette 2000/41)

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU
MC NL PT SE

(30) Priority: 30.03.1999 JP 8908499
06.04.1999 JP 9945799
09.04.1999 JP 10269499
14.04.1999 JP 10721299

(71) Applicant: MATSUSHITA ELECTRIC INDUSTRIAL CO., LTD.
Kadoma-shi, Osaka 571-8501 (JP)

(72) Inventors:
- MATSUMI, Chiyoko
  Suita-shi, Osaka 565-0862 (JP)
- YOSHIDA, Junji
  Neyagawa-shi, Osaka 572-0038 (JP)
- YAMADA, Masazumi
  Osaka-shi, Osaka 543-0071 (JP)
- KURANO, Yukio
  Higashi-osaka-shi, Osaka 579-8051 (JP)

(74) Representative:
Grünecker, Kinkeldey, Stockmair &
Schwanhäusser Anwaltssozietät
Maximilianstrasse 58
80538 München (DE)

(54) DATA PROCESSING SYSTEM, DATA TRANSMITTING/RECEIVING DEVICE, AND RECORDED MEDIUM

(57) A data processing system characterized in comprising:

a real-time interface driver which processes data of a device which comprises an interface which is required to ensure real-time processing during recording and/or reproducing;
a non real-time interface driver which processes data of a device which comprises an interface which is not required to ensure real-time processing during recording and/or reproducing:
an IFS manager which allocates instructions for predetermined processing regarding a file system to each one of control means which are to handle the instructions;
first control means which, using first file management information with respect to data which are in a first medium which is housed in the device which is required to ensure said-real-time processing, converts an instruction for said predetermined processing inputted through said IFS manager into a data processing instruction on said first medium and sends to said real-time interface driver: and
second control means which. using second file management information with respect to data which are in a second medium which is housed in the device which is not required to ensure said real-time processing, executes processing for accessing to a file on said second medium in accordance with the instruction for said predetermined processing inputted through said IFS manager and sends the instruction for said predetermined processing to said non real-time interface driver via an I/O sub system.

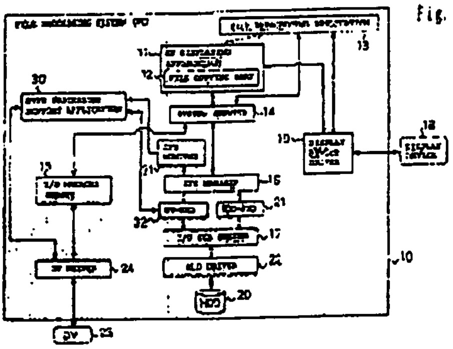


Fig. 1

Printed by Jouve, 75001 PARIS (FR)

Description

FIELD OF THE INVENTION

[0001] The present invention relates to a file processing system, a file processing method and a program recording medium for displaying file management information which denotes in what manner a file is recorded on a recording medium or for processing a file which is designated by the file management information.

[0002] The present invention also relates to a sending/receiving apparatus and a sending/receivingmethod for processing while treating stream data, which are recorded in a stream data recording/reproducing apparatus which performs real-time transmission, as a file on appearance.

[0003] The present invention further relates to an apparatus for and a method of transmission which realize real-time transmission of stream data and a sending/receiving apparatus and a sending/receiving method for transmitting data which contain copyright information.

BACKGROUND ART

[0004] Currently available as recording apparatuses for recording data in a personal computer (hereinafter referred to as "PC") include floppy disk drives (hereinafter referred to as "FDD"), hard disk drives (hereinafter referred to as "HDD") and CD-ROM units, etc. These recording apparatuses are connected with PCs via interfaces called IDE, SCSI, etc.

[0005] Operations of such a recording apparatus which is connected with a PC, hardware inside the PC, etc., are controlled by software called "Operating system (hereinafter referred to as "OS")" which is represented by Windows, MacOS and UNIX. Introduced in the OS is a concept of file system, for the purpose of handling data recorded on a recording medium of each recording apparatus, such as an FDD and a CD-ROM unit, as a file. A file system is a system in which each piece of data partitioned as a file bears information, such as a file name, a location at which the file data are recorded, a file length, a recording date/hour and the like, to thereby organize and manage such that data are treated as a file.

[0006] For example, a Windows PC is formed by hardware, kernel mode software serving as the OS anduser-mode software serving as applications. The OS known as Windows98 uses a file system called "FAT (File Allocation Table)" for an FDD and an HDD and a file system called "CDFS (CD File System)" for a CD-ROM unit. Software which controls the file systems is a file system driver (hereinafter referred to as "FSD"), and the FDS is contained in the OS. Only through the FSD, an access can be made to data (file), e.g., reading from a medium in the recording apparatus and writing recorded data (file).

[0007] By the way, digital interfaces based on new standards such as IEEE1394 Standard have been introduced to PCs, which has made it possible to connect a PC with equipment such as a VCR (for instance, DV (Digital Video Cassette)) which records and reproduces video and speech.

[0008] However, since DVs are not compatible with file systems, even when a DV is connected with a PC, a problem remains that the PC can not treat a content which is recorded in the DV as a file, and therefore, a file system for the DV (DV File system, abbreviated as DVFS) and a file system driver for the PC to handle the DVFS are necessary.

[0009] Further, even with the file system introduced to the DV, an application writes data in a file through the FSD by means of the following or repetition of the same

(1) File OPEN instruction - designation of a file name
(2) WRITE instruction to write in the opened file - designation of a write start position, a write data size and data-to-write
(3) File CLOSE instruction

The application and the OS determine the order of writing, the write size and the write timing, which is viewed from the DV as an asynchronous random request. In a similar manner, an application reads data from a file through the FSD by means of the following or repetition of the same:

(1) File OPEN instruction - designation of a file name
(2) READ instruction to read from the opened file - designation of a read start position and a read data size
(3) File CLOSE instruction

The application and the OS determine the order of reading, the read size and the read timing, which is viewed from the DV as an asynchronous random request

[0010] Sending and receiving stream data, the DV has a problem that it is impossible to respond to an asynchronous random access described above. Hence, a mechanism for responding to such an access is needed.

[0011] An action of switching operations little by little is not a premise for a magnetic tape recording machine of the sequential type such as a DV, and hence, even if the mechanism operates minutely to respond to a random access, as an operation mode for the mechanism is changed more frequently for which processing takes long, the mechanism and a tape get damaged. In addition, while such a random access is necessary to execute special processing such as editing, if the purpose is only to reproduce and display or copy AV data in real time, using stream data all at a time is more efficient. It is also necessary to appropriately accommodate an application for accessing asynchronously and at random and an application for handling stream data all at once.

[0012] On the other hand, for the purpose of connecting PCs with external equipment via IEEE1394 I/Fs, a WDM driver based on the WDM (Windows Driver Model) architecture has been introduced for Windows98 as a new driver which complies with IEEE1394 which is different from existing architectures, which has created an environment called DirectShow for handling stream data, such as video/audio data, on a PC. Although fusion of PCs and digital video/audio equipment, such as DVs, has been promoted in this manner, since PCs have never initially considered handling synchronous data, the current mechanisms can not handle real-time data without a break.

[0013] In other words, while sending from a PC to a DV demands to send a fixed amount of data for each video frame such that predetermined frame intervals are maintained, in the case that a part or all of processing is realized by software on the PC, processing of one frame does not complete in a constant period of time. Although a PC has a mechanism to instruct a processing time for processing data per unit, if a certain frame has used a long processing time, the subsequent instruction is to delay a processing time of the next frame rather than giving a higher priority to processing of the next frame. Hence, it is not possible to keep sending while maintaining predetermined frame intervals.

[0014] A number of digital video/audio apparatuses, including not only DVs but also set top boxes for receiving MPEG2-TP data (hereinafter referred to as "STB"), are capable of transferring data with other equipment by means of IEEE1394 I/Fs.

[0015] Further, from a standpoint of protection of copyright, for transmission of AV data, a scheme of transmitting copy right information on a header of an IEEE1394 transmission packet was proposed in "5C Digital Transmission Content Protection White Paper" in July, 1998. Since a current PC does not have a mechanism to deal with copy right information regarding AV data, a new mechanism to handle AV data in accordance with copy right information is necessary to send and receive AV data between PCs and other AV equipment.

SUMMARY OF THE INVENTION

[0016] There currently is a problem that a DV, sending and receiving stream data, can not respond to an asynchronous random access. Accordingly, the object of the present invention is to solve the problem that a mechanism for responding to such an access is necessary.

[0017] A further object of the present invention is to solve the problem that despite the recent ongoing fusion of PCs and digital video/audio equipment such as DVs, since PCs have never initially considered handling synchronous data, the current mechanisms can not handle real-time data without a break.

[0018] A still further object of the present invention is to solve the problem that a new mechanism to handle AV data in accordance with copy right information is necessary to send and receive AV data between PCs and other AV equipment.

[0019] The present invention (corresponding to claim 1) is a data processing system comprising an operating system which contains a file system driver and a device driver,

characterized in comprising:

a real-time interface driver which processes data of a device which comprises an interface which is required to ensure real-time processing during recording and/or reproducing;

a non real-time interface driver which processes data of a device which comprises an interface which is not required to ensure real-time processing during recording and/or reproducing;

an IFS manager which allocates instructions for predetermined processing regarding a file system to each one of control means which are to handle the instructions;

first control means which, using first file management information with respect to data which are in a first medium which is housed in the device which comprises an interface which is required to ensure said real-time processing, converts an instruction for said predetermined processing inputted through said IFS manager into a data processing instruction on said first medium and sends to said real-time interface driver; and

second control means which, using second file management information with respect to data which are in a second medium which is housed in the device which comprises an interface which is not required to ensure said real-time processing, executes processing for accessing to a file on said second medium in accordance with the instruction for said predetermined processing inputted through said IFS manager and sends the instruction for said predetermined processing to said non real-time interface driver via an I/O sub system.

[0020] The present invention (corresponding to claim 8) is a data processing system comprising an operating system which contains a file system driver and a device driver,

characterized in comprising:

display application means which displays files which can be handled in a file system, selects a desired file from the displayed files, and starts processing which is associated with said selected file;

a first processing part which executes processing on a first medium which stores data of a device which comprises an interface which is required, to ensure real-time processing during recording and/or reproducing;

a second processing part which executes processing on a second medium which stores data of a device which comprises an interface which is not required to ensure real-time processing during recording and/or reproducing:

device identifying means which identifies a device in which said selected file exists: and

a file starter part which drives said first or said second processing part in accordance with the identification result made by said device identifying means.

[0021] The present invention (corresponding to claim 9) is a data processing system comprising an operating system which contains a file system driver and a device driver,

characterized in comprising:

a real-time interface driver which processes data of a device which comprises an interface which is required to ensure real-time processing during recording and/or reproducing:

a non real-time interface driver which processes data of a device which comprises an interface which is not required to ensure real-time processing during recording and/or reproducing:

an IFS manager which allocates instructions for predetermined processing regarding a file system to each one of control means which are to handle the instructions;

first control means which, using first file management information with respect to data of the device which comprises an interface which is required to ensure said real-time processing which are stored in a first medium, converts an instruction for said predetermined processing inputted through said IFS manager into a data processing instruction on said first medium and sends to said real-time interface driver; and

second control means which, using second file management information with respect to data of the device which comprises an interface which is not required to ensure said real-time processing which are stored in a second medium, executes processing for accessing to a file on said second medium in accordance with the instruction for said predetermined processing inputted through said IFS manager and sends the instruction for said predetermined processing to said non real-time interface driver via an I/O sub system:

display application means which displays files which can be handled in a file system, selects a desired file from the displayed files, and starts processing which is associated with said selected file;

a first processing part which executes processing on a first medium which stores data of a device which comprises an interface which is required to ensure real-time processing during recording and/or reproducing:

a second processing part which executes processing on a second medium which stores data of a device which comprises an interface which is not required to ensure real-time processing during recording and/or reproducing:

drive identifying means which identifies a drive in which said selected file exists: and

a file starter part which drives said first or said second processing part in accordance with the identification result made by said device identifying means.

[0022] The present invention (corresponding to claim 13) is a data transmission apparatus, comprising:

time intervals generating means which creates predetermined time of day information:

outputting means which outputs a series of data frames as stream data;

sending means which divides said series of data frames forming said stream into packet data and sends: and

transmission managing means which manages said outputting means and said sending means,

characterized in that said transmission managing means enters said stream data to said sending means based on said predetermined time of day information.

[0023] The present invention (corresponding to claim 22) is a data transmission apparatus, comprising:

data conversion means which divides inputted stream data and adds header information to each piece to thereby obtain packets, and inserts packet processing start time information which is for the data receiving side into the header information of at least first packet of each frame of said stream data and thereafter outputs; and

an interface which, using a clock, outputs to a bus the packets processed by said data conversion means, at the transmission start time which corresponds to said packet processing start time information,

characterized in that the packet processing start time information of said packets is expressed by:

$$T1 = X + Z + Y (N - 1)$$

(where $X>=0, Z>=0$) assuming that X denotes the transmission start time for the first packet of the first frame, N denotes a frame number, Y denotes a frame period, Z denotes an initial value, and T1 denotes the processing start time of said packets.

[0024] The present invention (corresponding to claim

27) is a receiving apparatus, comprising:

an interface which receives a transmission packet which contains a transmission path header in which additional information is described and data blocks;
transmission path header separator means which separates said transmission packet into said transmission path header and said data blocks;
additional information extracting means which extracts said additional information from said transmission path header;
data packet generating means which generates from said data blocks a data packet which is obtained by combining one or more of said data blocks; and
additional information inserting means which adds or inserts said additional information to said data packet and outputs as an output packet of one type of data format which can be be processed by an application simultaneously,

characterized in that said application processes the output packet outputted from said additional information inserting means.

[0025] The present invention (corresponding to claim 31) is a receiving apparatus, comprising:

an interface which receives a transmission packet which contains a transmission path header in which additional information is described and data blocks;
transmission path header separator means which separates said transmission packet into said transmission path header and said data blocks;
additional information extracting means which extracts said additional information from said transmission path header;
data packet generating means which generates from said data blocks a data packet which is obtained by combining, one or more of said data blocks;
source packet header separator means which separates said data packet into a source packet header and a source packet; and
additional information inserting means which adds or inserts said additional information to said source packet or replaces said additional information, and outputs as an output packet of one type of data format which can be be processed by an application simultaneously,

characterized in that said application processes the output packet outputted from said additional information inserting means.

[0026] The present invention (corresponding to claim 38) is a data sending apparatus, comprising:

additional information separator means which, upon receipt of an input packet of one type of data format which can be processed by an application simultaneously, retrieves said additional information which is added or inserted to a data packet which forms said input packet;
data block generating means which generates one or more data blocks from said data packet;
transmission packet generating means which executes predetermined processing on said data blocks, inserts said additional information at a predetermined location in said data blocks, and accordingly generates said transmission packet; and
an interface which sends said transmission packet.

[0027] The present invention (corresponding to claim 42) is a sending apparatus, comprising:

additional information separator means which, upon receipt of an input packet of one type of data format which can be processed by an application simultaneously, constructs said input packet and retrieves said additional information from a transport packet which includes added, inserted or replaced additional information;
data packet generating means which combines a source packet header with said transport packet to thereby generate a data packet;
data block generating means which generates one or more data blocks from said data packet;
transmission packet generating means which executes predetermined processing on said data blocks, inserts said additional information at a predetermined location in said data blocks, and accordingly generates said transmission packet; and
an interface which outputs said transmission packet.

BRIEF DESCRIPTION OF THE DRAWINGS

[0028]

Fig. 1 is a block diagram of a file processing system according to a first preferred embodiment;
Fig. 2 is a block diagram of a file processing system according to a second preferred embodiment;
Fig. 3 is a block diagram of an FSD for a DV (used in the first and the second preferred embodiments) which is a third preferred embodiment;
Fig. 4 is a block diagram showing a first structure example of a DVFS processing support application (used in the first and the second preferred embodiments) which is a fourth preferred embodiment;
Fig. 5 is a block diagram showing a second structure example of a DVFS processing support application (used in the first and the second preferred embodiments) which is a fifth preferred embodiment;
Fig. 6 is a block diagram of a data transmission apparatus according to a sixth preferred embodiment;

Fig. 7 is an explanatory diagram showing time intervals information used by the data transmission apparatus according to the sixth preferred embodiment;

Fig. 8 is a block diagram of a data transmission apparatus according to a seventh preferred embodiment;

Fig. 9 is an explanatory diagram showing a method of transmitting DV frame data on an IEEE1394 bus;

Fig. 10 is a timing chart showing the timing of sending and receiving isochronous packets on an IEEE1394 bus:

Fig. 11 is an explanatory diagram showing a method of calculating SYT;

Fig. 12 is a flow chart showing an example of operations of a CIP generating part 115;

Fig. 13 is a block diagram of a receiving apparatus according to an eighth preferred embodiment;

Fig. 14 is a block diagram of a receiving apparatus according to a ninth preferred embodiment;

Fig. 15 is a block diagram of a receiving apparatus according to a tenth preferred embodiment;

Fig. 16 is a block diagram of a receiving apparatus according to an eleventh preferred embodiment;

Fig. 17 is an explanatory diagram showing a structure example of an input/output packet which is used in the eighth, the ninth, the tenth and the eleventh preferred embodiments;

Fig. 18 is an explanatory diagram showing a method of transmitting an MPEG2 TS packet on an IEEE1394 bus: and

Fig 19 is an explanatory diagram of what is realized by a program which is a twelfth preferred embodiment.

BEST EMBODIMENT FOR IMPLEMENTING THE INVENTION

[0029] In the following, preferred embodiments of the present invention will be described with reference to associated drawings.

(First Preferred Embodiment)

[0030] First a structure of a file processing system according to a first preferred embodiment of the present invention will be described with reference to Fig. 1. Fig. 1 is a block diagram of the file processing system according to the first preferred embodiment. The following assumes that Windows PCs realize file processing systems. In Fig. 1, denoted at 10 is a PC (file processing system), denoted at 11 is an FS displaying application, denoted at 12 is a file copying application, denoted at 13 is a file reproducing application, denoted at 14 is a system service, denoted at 15 is an I/O manager subset, denoted at 16 is an IFS manager, denoted at 17 is an I/O sub system, denoted at 18 is a displaying device, denoted at 19 is a displaying device driver, denoted at 20 is a WDM-incompatible device (described as an HDD as an example in this embodiment), denoted at 21 is an HDD file system driver (HDD-FSD), denoted at 22 is an HDD driver, denoted at 23 is a WDM-compatible device (described as a DV as an example in this embodiment), and denoted at 24 is a DV driver, all of which are elements which already are present within conventional PCs. Denoted at 30 is a DVFS support application, denoted at 31 an IFS monitor, and denoted at 32 is a DV file system driver (DV-FSD).

[0031] Without the DVFS support application 30, the IFS monitor 31 and the DV-FSD 32, the file processing system 10 is merely a system which can handle files which are recorded in the HDD 20. Recorded files and a directory structure on the HDD 20 are managed by the HDD-FSD 21 based on an FAT (File Allocation Table).

[0032] The DV 23 houses a magnetic tape, and data can be recorded in the form of a file on the magnetic tape. File management information which is necessary to manage recorded files, a directory structure and the like is also recorded. The file management information may include tape-related information such as tape IDs and volume names, file entries each set for each file, directory entries each set for each directory, etc. Each entry consists of information such as a file name and a creation date, and a file entry further includes information such as a file size and a recording location on the tape. A file size may be a frame count or a byte count (or may be controlled in the both figures).

[0033] Now, the following will describe an operation in a case that a recorded content on the magnetic tape inside the DV 23 is displayedbased on the associated file management information in the structure described above.

[0034] The description below assumes that a content recorded on the magnetic tape inside the DV 23 is displayed using the FS displaying application 11 which is used to display contents of files which are recorded on the HDD 20, an FDD, a CD-ROM or the like in the PC 10

[0035] To obtain information regarding a desired file, first, the FS displaying application 11 sends an OPEN request to open that file to the DV-FSD 32 through the system service 14 and the IFS manager 16. After the DV-FSD 32 accesses the file management information and confirms the existence of the file and returns, the FS displaying application 11 sends an acquisition request for information such as a file size, a file update date/hour or the like to the DV-FSD 32 through the system service 14 and the IFS manager 16, and the DV-FSD 32 accordingly accesses the file management information, extracts the demanded information and returns the information. Upon acquisition of the desired information, the FS displaying application 11 sends a CLOSE request to close the file to the DV-FSD 32 through the system service 14 and the IFS manager 16, and the DV-FSD 32 accordingly discards the information held with respect to the file.

[0036] With the processing as described above repeated, one can look at a displayed list of recorded files using the FS displaying application 11 such as Explorer in Windows available from Microsoft Corporation.

[0037] Next, an operation in a case that a file in the DV 23 is copied to the HDD 20 will be described. First, a user selects a file in the DV 23 which is to copy from a list which is displayed by the FS displaying application 11, and instructs any desired directory on the HDD 20 as a copy destination. The file copying part 12 sends to the IFS manager 16 through the system service 14 information regarding the selected file and a copy request to the HDD 20.

[0038] More specifically, in this case, the request sent to the IFS manager 16 is a notice of starting processing a file (which is on the DV 23 or the HDD 20) (hereinafter referred to as "OPEN-Request"), a request to read data of a file (hereinafter referred to as "READ-Request"), a request to write data in a file (hereinafter referred to as "WRITE-Request"), or a notice of terminating processing on a file (hereinafter referred to as "CLOSE-Request").

[0039] An OPEN-Request is sent together with information regarding a file to designate, such as a file name, a device number and a file ID. A READ-Request and a WRITE-Request are sent together with information such as a file ID and a location and the size of demanded data in a file. A CLOSE-Request is sent together with a file ID.

[0040] The IFS manager 16 sends to the DV-FSD 32 an OPEN-Request regarding a file which is on the DV 23 selected as a copy source, thereafter sends a READ-Request for data, and then sends a CLOSE-Request after reading all data. Alternatively, the IFS manager 16 sends to the HDD-FSD 22 an OPEN-Request regarding the HDD 20 which is selected as a copy destination, thereafter sends a WRITE-Request for data, and then sends a CLOSE-Request after writing all data. In either case, a file name at the copy source and a file name at the copy destination are different when viewed as a full path name but the same as a file name excluding a directory-related portion.

[0041] The IFS monitor 31 has a function of monitoring an output from the IFS manager 16 and sending to the DVFS support application 30 (and may further have a function of discarding information which the DVFS support application 30 does not need). Hence, the IFS monitor 31 acquires the OPEN-Request, the READ-Request and the CLOSE-Request described above directed to the DV 23 and the OPEN-Request, the WRITE-Request and the CLOSE-Request described above directed to the HDD 20, and sends them to the DVFS support application 30.

[0042] The DV-FSD 32 extracts corresponding file management information based on the file name sent together with the OPEN-Request, and sends the extracted file management information together with each Request to the DVFS support application 30.

[0043] The DVFS support application 30 checks requests sent from the IFS monitor 31, and if there is a write request demanding a file on the HDD 20 which corresponds to a data read request demanding a file on the DV 23, the DVFS support application 30 determines that there is a request to copy the file on the DV 23 to the HDD 20 and executes processing of capturing, to the HDD 20 in real time, a section of DV stream data which are obtainable through the DV driver 24 from the DV 23, starting at a portion which is designated as the beginning of the file up to a portion which is designated as the end of the file.

[0044] Meanwhile, in the case of copying a file on the HDD 20 to the DV 23, in a manner opposite to the above, the IFS manager 16 sends a WRITE-Request to the DV-FSD 32 but a READ-Request to the HDD-FSD 22, so that the DVFS support application 30 performs processing of sending a file on the HDD 20 to the DV 23 and thereafter recording on the magnetic tape in the DV 23 starting at a portion which is designated as a record start position.

[0045] The processing above makes it possible to copy from the DV 23 to the HDD 20 and from the HDD 20 to the DV 23 file by file based on an asynchronous random request, even if the DV 23 is capable of only receiving and outputting stream data.

(Second Preferred Embodiment)

[0046] Next, a structure of a file processing system according to a second preferred embodiment of the present invention will be described with reference to Fig. 2. Fig. 2 is a block diagram of the file processing system according to the second preferred embodiment which is the same as the first preferred embodiment in that a Windows PC realizes the system. In Fig. 2, the structure is mostly the same as that shown in Fig. 1, and denoted at 40 is a file reproduce/starter application, denoted at 41 is a drive identifying part, denoted at 42 is an HDD file starter, and denoted at 43 is a DV file starter.

[0047] Now, the following will describe an operation in a case that a file recorded on a magnetic tape in the DV 23 is reproduced and displayed based on the associated file management information in the structure described above.

[0048] A mechanism is that as a file is selected and double-clicked in a file list provided by the FS displaying application 11, an application which is designated depending on the type of a file is selected and started, and the started application executes an operation of "opening" the selected file. In general, if the file represents AV data, the file reproducing application 13 is started. If it is on the HDD 20, with the file reproducing application 13, real-time reproduction is possible (This however bears a possibility that a video signal misses some frames.). When one wishes to "open" a file on the DV 23 as in the case of opening a file on the HDD 20, to reproduce and display AV data recorded in the DV 23 in real time is very difficult, and therefore, a dedicated application is necessary.

[0049] An application which is designated for an AV data file is not the file reproducing application 13 but is the file reproduce/starter application 40. As the file reproduce/starter application 40 is started, the drive identifying part 41 judges on which drive a selected file is present. Determining that it is a drive other than the DV, the drive identifying part 41 notifies the file reproducing application 13 of information of the selected file through the HDD file starter 42 and starts the file reproducing application 13 so that an operation of "opening" the selected file will be executed. When the drive identifying part 41 judges that it is a DV file, information of the selected file is notified to the DVFS support application 30 through the DV file starter 43, and the DVFS support application 30 executes processing of reproducing and displaying, in real time, DV stream data which are obtainable through the DV driver 24 from the DV 23.

[0050] The processing above makes it possible to reproduce and display data in the DV 23 file by file based on an asynchronous random request, even if the DV 23 is capable of only outputting stream data.

[0051] Although the foregoing has described that data are video/speech data and the applications are for reproducing and displaying video/speech data according to the present invention, even in the case of data of other types, a similar effect is obtainable using applications which correspond to the other type data.

(Third Preferred Embodiment)

[0052] A structure of the DV-FSD 32 which is used in the first and the second preferred embodiments of the present invention will be described with reference to Fig. 3. Fig. 3 is a block diagram of the DV-FSD 32, wherein denoted at 50 is a request identifying/return sending part, denoted at 51 is a file entry extracting part, denoted at 52 is a file management information holding part, denoted at 53 is an access file entry holding part, denoted at 54 is a first file information extracting part, denoted at 55 is a second file information extracting part.

[0053] The DV-FSD having such a structure operates in the following manner. The request identifying/return sending part 50, being an interface with the IFS manager 16, receives a request and returns information/data demanded by the request and a return code. Further, the request identifying/return sending part 50 discriminates requests of a plurality of types.

[0054] When a request is OPEN, based on a file name received together with the request, the file entry extracting part 51 extracts a file entry which corresponds to the file name from the file management information holding part 52, registers in the access file entry holding part 53, and sends the OPEN request to the DVFS support application 31. OPEN requests are notifications to start processing corresponding files. In general, the system handles a file in accordance with only a file ID and associates a file ID with a file name only when receiving an OPEN request, and therefore, it is necessary for the access file entry holding part 53 to associate a file ID with a file entry. From information held by the file management information holding part 52, in the absence of a file which corresponds to the designated file name, the request identifying/return sending part 50 returns a return code which is indicative of a failure.

[0055] When the request is READ or WRITE, the second file information extracting part 55 extracts information regarding the designated file from the access file entry holding part 53 based on the file ID received together with the request, adds such information to the request and sends to the DVFS support application 30. In the case that the DVFS support application 30 determines that copying of a file in the DV 23 to the HDD 20 or copying a file on the HDD 20 to the DV 23 is demanded, data processing in response to the READ or WRITE request is not executed, and instead, the request identifying/return sending part 50 returns a return code which is indicative of a success.

[0056] When the request is CLOSE, the second file information extracting part 55 extracts information regarding the designated file from the access file entry holding part 53 based on the file ID received together with the request and sends the request to the DVFS support application 30, and the request identifying/return sending part 50 returns a return code which is indicative of a success.

[0057] For a request other than the above, the request needs not be sent to the DVFS support application 30, and therefore, the first file information extracting part 54 extracts information regarding the designated file from the access file entry holding part 53 based on the file ID received together with the request and the request identifying/return sending part 50 sends it together with a success return code to the IFS manager 16.

[0058] Owing to the DV-FSD 32 having such a structure described above, even if the DV 23 is capable of only receiving and outputting stream data, it is possible to execute processing file by file based on an asynchronous random request.

(Fourth Preferred Embodiment)

[0059] A structure of the DVFS support application 30 which is used in the first and the second preferred embodiments of the present invention will be described with reference to Fig. 4. Fig. 4 is a block diagram of the DVFS support application 30, wherein denoted at 60 is a supportprocessing judging part, denoted at 61 is a reproducing/displaying part, denoted at 62 is a DV→HDD copying part, denoted at 63 is an HDD→DV copying part, and denoted at 64 is a file management information updating/acquiring part.

[0060] The DVFS support application having such a structure operates in the following manner. The support processing judging part 60 judges processing regarding the DV 23 based on a request from the DV-FSD 32 and information from the IFS monitor 31, and instructs to execute appropriate processing.

[0061] When instructed to reproduce a file which is recorded on the tape in the DV 23, the reproducing/displaying part 61 is started by the file reproduce/starter application 40 and executes reproducing/displaying processing while working in cooperation with the DV driver 24. More specifically, the following is carried out as a series of operations:

(1) An instruction to search for a front part of a file is sent to the DV 23 through the DV driver 24.
(2) An instruction for reproduction is sent to the DV 23 through the DV driver 24.
(3) Data reproduced by the DV 23 and sent from the DV 23 through the DV driver 24 are displayed.
(4) When a last part of the file is reached, an instruction to stop is sent to the DV 23 through the DV driver 24.

[0062] When instructed to copy a file recorded on the tape in the DV 23 to the HDD 20, the DV→HDD copying part 62 is started in accordance with the judgment made by the support processing judging part 60 and executes copying while working in cooperation with the DV driver 24. In short, the following is carried out as a series of operations:

(1) An instruction to search for a front part of a file is sent to the DV 23 through the DV driver 24.
(2) An instruction for reproduction is sent to the DV 23 through the DV driver 24.
(3) Data reproduced by the DV 23 and sent from the DV 23 through the DV driver 24 are written on the HDD 20.
(4) When a last part of the file is reached, an instruction to stop is sent to the DV 23 through the DV driver 24.

[0063] When instructed to copy a file on the HDD 20 to a tape in the DV 23, the HDD→DV copying part 63 is started in accordance with the judgment made by the support processing judging part 60 and executes copying while working in cooperation with the DV driver 24. In short, the following is carried out as a series of operations:

(1) An instruction to search for the most recent file recording end position in the tape is sent to the DV 23 through the DV driver 24.
(2) Data to be read from the HDD 20 are sent from the DV 23 through the DV driver 24
(3) An instruction for recording is sent to the DV 23 through the DV driver 24.
(4) When a last part of the file is reached, an instruction to stop is sent to the DV 23 through the DV driver 24.

[0064] Since the support processing judging part 60 instructs the DV→HDD copying part 62 and the HDD→DV copying part 63 to carry out an operation file by file, at least a recording location and the size of the file which are information regarding the file needs be provided together with the instruction for operation. If a file name as well is to be displayed when a message indicating that the processing is ongoing is displayed on a screen, information regarding the file name as well needs be provided.

[0065] When the IFS manager 16 and the other parts demand information contained in the file management information, accessing a tape each time takes very long time and imposes a very large burden upon the mechanism of the DV 23 and the tape itself, and hence, it is preferable that the file management information holding part 52 within the file processing system 10 holds such information in advance and the file management information holding part 52 is accessed usually.

[0066] Noting this, the file management information updating/acquiring part 64 executes an operation of recording the file management information held by the file management information holding part 52 on a tape or making the file management information holding part 52 acquire the file management information recorded on the tape.

[0067] In order to prevent the file recorded on the tape, the file management information recorded on the tape and the file management information held by the file management information holding part 52 from becoming different from each other, it is necessary to record the file management information as well on the tape at least immediately after recording the file on the tape. (When it is known from the beginning that more than one files are to be recorded subsequently, the file management information may be recorded on the tape immediately after recording the last file on the tape.)

[0068] Upon start-up of the file processing system 10, it is necessary to acquire the file management information. Further, since the mechanism of the DV does not require the DV driver 24 to transmit this information to the file processing system 10 even despite exchange of a medium, and therefore, it is desirable that the file management information updating/acquiring part 64 has a function of checking whether a medium has been exchanged or not.

[0069] With a medium ID included in the file management information, whether the file management information in the file management information holding part 52 and the file management information acquired by the file management information updating/acquiring part 64 are on the same medium can be determined based on a medium ID, and hence, it is possible to judge whether a medium has been exchanged or not.

[0070] In addition, with the most recent updating date/time information included in the file management information, which one of the file management information in the file management information holding part 52 and the file management information acquired by the file management information updating/acquiring part 64 is the most recent file management information can be found, which in turn makes it possible to deal with a situation that the two pieces of file management information do not match with each other.

[0071] Other than the start-up of the file processing system 10, the timing of acquiring the file management information may be issuance of a particular instruction from the FS displaying application 11, for example (on the condition that the DV driver 24 is not handling data as during copying or reproducing). Alternatively, acquisition may be on a regular basis (on the condition that the DV driver 24 is not handling data as during copying or reproducing).

[0072] By means of the DVFS support application 30 which works in cooperation with the DV-FSD 32 having such a structure described above, even if the DV 23 is capable of only receiving and outputting stream data, it is possible to execute processing file by file based on an asynchronous random request.

[0073] Although the foregoing has described that the most recent updating date/time information and medium IDs are a part of the file management information, medium IDs may be stored in an auxiliary memory which is attached to a tape. Alternatively, since the values of medium IDs will not be changed once specific values are set with respect to the tape, the medium IDs may be recorded on a label of a package or in an auxiliary data recording area of a track formed on the tape, which enables to quickly acquire these values without searching for the file management information.

[0074] In short, the medium IDs may be recorded in any place, such as on the tape, in an auxiliary memory or on a package, as far as they can be acquired by the DV driver 24.

[0075] While the foregoing has described the support processing judging part 60 as included in the DVFS support application 30 in relation to the third and the fourth preferred embodiments, a similar effect is obtainable if this function is assigned to the DV-FSD 32 (on the condition that the DV-FSD 32 receives an output from the DVFS support application 30).

[0076] Although the foregoing has described a sequence of successful processing, when an error occurs during processing, each means operates in accordance with the error.

[0077] Wherever the file management information for the DVFS which the FSD refers to is, may it be on the HDD, in a memory or somewhere on the tape, the present invention remains applicable. If the system is for handling a plurality of tapes in the same PC, with medium IDs which associate the tapes with the file management information registered in the filemanagement information, the management information needs not be recorded on the tapes.

[0078] Further, where the file management information is to be recorded on a tape, the system may be such a system that keeps the information uploaded on the HDD or in a memory and access to the HDD or memory in reality.

[0079] It is also possible to handle general data, not only video and speech data, in a similar fashion.

[0080] Each processing of operation is merely an example, and thus, may include a function of ± fast forwarding, ± slow or the like in addition to regular reproducing operations such as search → reproduce → stop

(Fifth Preferred Embodiment)

[0081] A second structure example of the DVFS support application which is used in the first and the second preferred embodiments of the present invention will be described with reference to Fig. 5. Fig. 5 is a block diagram of a DVFS support application 30a, wherein denoted at 65 is a support processing judging part, denoted at 61 is a reproducing/displaying part, denoted at 62 is a DV→HDD copying part, denoted at 63 is an HDD→DV copying part, denoted at 64 is a file management information updating/acquiring part, denoted at 66 is a random access processing part, denoted at 70 is a request analysis part, denoted at 71 is a buffer managing part, denoted at 72 is a buffer, and denoted at 73 is a DV control part. The reproducing/displaying part 61, the DV→HDD copying part 62, the HDD→DV copying part 63 and the file management information updating/acquiring part 64 are the same as those of the fourth preferred embodiment. The supportprocessing judging part 65 has a random access function in addition to the function of the support processing judging part 60 of the fourth preferred embodiment.

[0082] The support processing judging part 65 feeds the random access processing part 66 with an OPEN-Request, a READ-Request, a WRITE-Request and a CLOSE-Request directed to the DV 23 other than a request which is determined as copying processing from the DV 23 to the HDD 20 and a request which is determined as copying processing from the HDD 20 to the DV 23. When feeding with an OPEN-Request or a CLOSE-Request, the support processing judging part 65 provides information regarding a designated file as well together with the request. When feeding with a READ-Request, the support processing judging part 65 provides, together with a designated manner of returning requested data, information which designates which portion of the file is requested, in addition to information regarding a recording location and the size of the designated file on the tape.

[0083] When feeding with a WRITE-Request, the support processing judging part 65 provides, together with requested data, information which designates which portion of the file is requested, in addition to information regarding a recording location and the size of the designated file on the tape. The requested data are designated by means of an offset byte count from the beginning of the file and a byte size of the requested data, both in the case of READ and WRITE.

[0084] In the random access processing part 66, first, the request analysis part 70 identifies a request.

[0085] When the request is a READ-Request, from the recording location of the file on the tape and and an offset byte count from the beginning of the file, the request analysis part 70 confirms which portion on the tape bears the requested data. The buffer managing part 71 confirms whether the requested data are in the buffer 72 or not. When the requested data are not in the buffer 72, the DV control part 73 sends to the DV 23 through the DV driver 24 an instruction for an operation of searching for a portion which bears the requested data, sends an instruction for reproduction to the DV 23 through the DV driver 24, and writes in the buffer 72 data which are reproduced by the DV 23 and received through the DV driver 24.

[0086] Meanwhile, the buffer managing part 71 transfers the requested data to the support processing judging part 65 from the buffer 72 via the request analysis part 70.

[0087] When the request is aWRITE-Request, from the recording location of the file on the tape and an offset byte count from the beginning of the file, the request analysis part 70 confirms which portion on the tape should bear the requested data, and more specifically, howmany bytes should be written from which byte in which frame. The buffer managing part 71 confirms whether the requested data are in the buffer 72 or not.

[0088] When the requested data are not in the buffer 72, the DV control part 73 sends to the DV 23 through the DV driver 24 an instruction for an operation of searching for a portion which should be bearing the requested data, sends an instruction for reproduction to the DV 23 through the DV driver 24, and writes in the buffer 72 data which are reproduced by the DV 23 and received through the DV driver 24. The buffer managing part 71 writes the data demanded by the WRITE-Request over data which are in the buffer 72, sends at least re-written frame data on the buffer 72 to the DV 23 through the DV driver 24, and over-writes in a designated portion on the tape which is in the DV 23.

[0089] While recording to the DV 23 proceeds frame by frame, since a WRITE-Request designates in the unit of bytes to write from somewhere in a frame, the procedure above is necessary. However, when the buffer managing part 71 determines from the processing up to that stage that necessary data are in the buffer 72, it is not necessary to reproduce data from the DV 23.

[0090] The structure described above makes it possible to deal with not only particular applications but general applications as well (such as real-time reproduction/displaying and copying) even if the DV 23 is capable of only receiving and outputting stream data, and therefore it is possible to appropriately deal with the particular applications and the general applications respectively.

[0091] Since a tape is a sequential medium, for the purpose of reproducing a file of video/speech data for instance in real time, one file must be recorded in continuous areas on a tape. When a file is to be over-written, as over-writing takes place on a medium, if a size exceeds that of a file as it is before over-writing, recording becomes impossible. Hence, it is necessary to execute appended-recording during an over-write save as well not only during a save of a new file.

[0092] In the case of a tape formatted in advance, similar processing to that for a WRITE-Request described above may be executed. In the case of a tape not formatted in which there is a frame bearing no record before a frame to record in, a dummy frame needs be recorded. Conversely, where a WRITE-Request has a precondition that data are specified in a sequential order starting with data at the beginning of a file, processing of reproducing data first in the buffer 72 from the DV 23 is not necessary.

[0093] Further, although the foregoing has described that data to be transmitted are DV data, data to be transmitted may be other data.

[0094] Further, although the foregoing has described that an I/F is an IEEE1394 interface, other I/F may be used.

[0095] Further, a receiving apparatus may be realized by software and/or hardware.

(Sixth Preferred Embodiment)

[0096] Next, a sixth preferred embodiment of the present invention will be described with reference to Fig. 6.

[0097] Fig. 6 is a block diagram of a data transmission apparatus. In Fig. 6, denoted at 101 is a manager, denoted at 102 is an HDD reader part, denoted at 103 is a DV sender part, denoted at 104 is a buffer, denoted at 105 is an HDD, denoted at 106 is a DV, denoted at 107 is a DV data file, and denoted at 108 is a time intervals information generating part.

[0098] The following assumes that the DV data file 107 is formed by video data of the NTSC system.

[0099] As transmission starts, first, the manager 101 sends an output instruction to the HDD reader part 102. Receiving the output instruction, the HDD reader part 102 reads frame data from the DV data file 107 recorded on the HDD 105, adds time intervals information generated by the time intervals information generating part 108 to this, and writes in the buffer 104 as transmission data. The time intervals information indicates the time of day to start processing frame data in the DV sender part 103 and the time of day to end processing frame data in the DV sender part 103.

[0100] Further, the manager 101 sends an input instruction to the DV sender part 103 based on the time intervals information written in the buffer 104, and when a free space is created in the buffer 104, the manager 101 sends the next output instruction to the HDD reader part 102 based on remaining area information regarding the buffer 104.

[0101] That is, the input instruction is sent to the DV sender part 103 such that the DV sender part 103 starts inputting frame data during an occupation period which is indicated by the time intervals information written in the buffer 104 and completes inputting the frame data. Receiving the input instruction, the DV sender part 103 reads the transmission data from the buffer 104, converts it into a transmission packet, and outputs it to the DV 106 via an IEEE1394 bus.

[0102] This is repeated, thereby transmitting data in the DV data file 107.

[0103] When a processing time of frame data becomes uneven because of other processing executed simultaneously by the PC or for other reason and actual processing becomes too slow, the time of day designated by the time intervals information becomes different from the actual time of day of processing. To reduce such a difference, the manager 101 creates a time intervals correction instruction based on load information regarding the DV sender part 103 and the time intervals information, and sends it to the time intervals information generating part 108.

[0104] For example, it is assumed that since a frame frequency of video data of the NTSC system is 30/1.001 (≒29.97) Hz, an occupation period of frame data is initially set to 0.033 sec. In response, the time intervals information regarding the 0-th frame data holds (the start time, the end time) = (0, 0.033). If the processing in the DV sender part 103 takes long, the manager 101 sends a time intervals correction instruction to the time intervals information generating part 108 so as to extend the occupation period of frame data.

[0105] However, for the purpose of maintaining frame intervals, the time intervals information generating part 108 does not receive the time intervals correction instruction sent from the manager 101 or receives but ignores it, and so that values shown in Fig 7 are used as the time intervals information corresponding to frame data which bear a frame number n. In other words, the start time is n*1.001/30 sec. and the end time is (n+1) *1.001/30 sec.

[0106] By means of the time intervals information generating part 108 having such a structure described above, frame data are supplied to the DV sender part 103 while maintaining the frame frequency intervals approximately, and as the data are converted into a predetermined packet and sent to the DV 106 in this condition, stable sending to the DV 106 is realized.

[0107] Although the start time is n*1.001/30 sec and the end time is (n+1) *1.001/30 sec in the foregoing, values close to these, such as these values with their fractions rounded off, may be used.

[0108] Further, although the foregoing has described that the DV data file 107 is formed by video data of the NTSC system, the DV data file 107 may be formed by other video data of a different method such as the PAL method or other stream data such as MPEG data.

[0109] Further, although the foregoing has described that the time intervals information according to this embodiment is set to define that the start time is n*1.001/30 sec and the end time is (n+1) *1.001/30 sec, this is not limiting. When the DV data file 107 is video data of other method except for the NTSC system or other stream such as MPEG data, the time intervals information may be determined based on the other method, a frame frequency of the stream, etc. In short, the time intervals information only needs be set so as to correspond to the timing which is needed by an apparatus which receives packet data of the data frame, such as the DV 106.

[0110] Further, although the foregoing has described that the manager 101 sends an input instruction to the DV sender part 103 based on the time intervals information written in the buffer 104, this is not limiting. The manager 101 may send an input instruction to the DV sender part 103 based on the time intervals information written in the buffer 104 and burden information regarding the DV sender part 103. In this case, the manager 101 fine-tunes the processing that the DV sender part 103 reads transmission data from the buffer 104 based on the burden information regarding the DV sender part 103. One example of a fine-tuning operation by the manager 103 (101 ???) will be described below.

[0111] For instance, when the DV sender part 103 can not read transmission data from the buffer 104 temporarily because of other processing executed simultaneously by the PC so that the DV sender part 103 can be destroyed if the manager 101 sends an input instruction to the DV sender part 103, the manager 103 learns from the burden information that the DV sender part 103 can not accept an input instruction and temporarily postpones sending of an input instruction to the DV sender part 103. As the DV sender part 103 becomes capable of normally accepting an input instruction, the manager 103 sends the postponed input instruction to the DV sender part 103. Further, to make up for the delay in the processing of the transmission data, time intervals for sending an input instruction are decreased. In this manner, the manager 101, considering not only the time intervals information but the burden information as well, sends the input instruction to the DV sender part 103 in such a manner that the processing completes during an occupation period which is indicated by the time intervals information while making sure that the DV sender part 103 will not break down. In this manner, the manager 103 may fine-tune the DV sender part 103 while considering not only the time intervals information but the burden information as well to ensure that the processing at the DV sender part 103 will not break down.

[0112] Further, although the foregoing has described that the manager 101 creates a correction instruction for correcting the time intervals information based on the burden information regarding the DV sender part 103 and the time intervals information and sends it to the time intervals information generating part 108 according to this embodiment, this is not limiting. The manager 101 may not create a correction instruction for correcting the time intervals information. Hence, the manager 101 does not send correction information to the time intervals information generating part 108. In this manner, an effect equivalent to that according to this embodiment is obtainable, and it is possible to realize a data transmission apparatus having a simpler structure.

[0113] Further, the time of day information according to the present invention is not limited to express, in the DV sender part 103, the time at which processing of frame data should start and the time at which the processing should end as in the case of the time intervals information according to this embodiment, but may express only the time at which processing of frame data should start. In this case, the time at which processing of the next frame data should start maybe the time at which the ongoing processing of the current frame data should end.

[0114] Further, the data transmission apparatus maybe formed by hardware or software, or the both.

[0115] Further, the time intervals information generating means 108 according to this embodiment is an example of the time intervals information generating part according to the present invention, the HDD reader part 102 according to this embodiment is an example of the outputting means according to the present invention, the DV sender part 103 according to this embodiment is an example of the sending means according to the present invention, the manager 101 according to this embodiment is an example of the transmission managing means according to the present invention. In addition, the time of day information according to this embodiment is an example of the time intervals information according to the present invention.

[0116] Further, the manager 101 according to this embodiment may be a part of DirectShow which manages data transmission, or a data transmission-managing portion of application software which can be handle stream data, such as video/speech data, on a PC.

[0117] Further, the data transmission apparatus maybe a data transmission apparatus which executes transmission of other types of data, such as to reproduce and display a DV data file, using a DV reproducing part instead of the DV sender part 103.

[0118] According to this embodiment, fluctuations in transmission of frame data are reduced so that it is possible to transmit stream data in a stable manner.

(Seventh Preferred Embodiment)

[0119] Next, a seventh preferred embodiment of the present invention will be described with reference to Fig. 8.

[0120] Fig. 8 shows an example of a data transmission apparatus according to the seventh preferred embodiment of the present invention. In Fig. 8, denoted at 111 is an application, denoted at 112 is a DV driver, denoted at 113 is the transmission start time information holding part, denoted at 114 is an SYT calculating part, denoted at 115 is a CIP generating part, denoted at 116 is at IEEE1394 I/F, denoted at 117 is a time counter on a bus, and denoted at 118 is a DV.

[0121] Further, Fig. 9 is an explanatory diagram showing a structure of DV frame data. As shown in Fig. 9 (a), DV data expressing video of the NTSC system consists of 1500 DIF blocks having the length of 80 bytes. A CIP (Common Isochronous Packet) shown in Fig. 9 (b) is collection of data which can be transmitted in one IEEE1394 cycle, and is formed by six DIF blocks and a CIP header. The last two bytes of the CIP header are information called "SYT" which expresses the time at which the DV 118 starts processing after receiving data using the lower two bytes of a time counter according to IEEE1394 shown in Fig. 9 (d). However, FFFFh may be substituted in an SYT area in the CIP except for at the beginning of a frame. It is an isochronous packet as that shown in Fig. 9 (c) that is actually transmitted, and the isochronous packet is formed by further adding a packet header, a header_CRC and a data_CRC to the CIP.

[0122] A time in IEEE1394 is formed by three types of counters called a "second_count," a "cycle count" and "cycle offset" as shown in Fig. 9 (d). The second_count has a width of seven bits and circulates among values from 0 to 127, and the count unit is one second. The cycle_count has a width of thirteen bits and circulates among values from 0 to 7999, and the count unit is 125 microseconds. The cycle_count has a width of twelve bits and circulates among values from 0 to 3071, and the count unit is 1/24576000 second.

[0123] The transmission processing is executed in the following manner. The application 111 outputs one frame of DV data to the DV driver 112 as frame data. The DV driver 112 divides the received frame data for the six DIF blocks, adds the CIP header and converts into a CIP, and outputs it to the IEEE1394 I/F 116. The IEEE1394 I/F 116 adds a packet header, a header_CRC and a data_CRC to the CIP received from the DV driver 112 to converts into an isochronous packet, and outputs to the DV 118.

[0124] Fig. 10 is a timing chart showing the timing of sending and receiving isochronous packets. The condition for the DV 118 to receive an isochronous packet and process normally inside is:

(1) The time TY designated by the SYT of the isochronous packet is after the time TD at which the DV 118 has received the isochronous packet; and

(2) A difference between the time TY designated by the SYT of the isochronous packet and the time TD at which the DV 118 has received the isochronous packet is 450 microseconds or less. That is, the difference must be TD < TY < TD + 450 microseconds.

[0125] To satisfy the condition above, the transmission start time information holding part 113 acquires the time of day information on the bus from the time counter on the bus which is in the IEEE1394 I/F 116, sets and holds the transmission start time TS based on the acquired time, and sends an instruction regarding the transmission start time for the entire stream data to the IEEE1394 I/F 116. The SYT calculating part 114 calculates the SYT for each frame based on the transmission start time TS set by the transmission start time information holding part 113, and the CIP generating part 115 creates a CIP of a predetermined format

[0126] Fig. 11 shows a method of calculating the SYT. Considering that a sending delay is a few dozens of microseconds or shorter and considering the relationship described above between TY and TD, an offset tso from TS of the time TY designated by the SYT of an isochronous packet which is at the beginning of stream data to be transmitted is set to 250 microseconds. As to the following frames, since this example is on a video signal of the NTSC system, the SYT is incremented by 1.001/30 second each time. In this manner, it is possible to provide an SYT which meets the condition above regarding all isochronous packets which are to be transmitted. If a calculated result has fractions, the fractions are rounded off. The SYT values correspond to an example where TS = 0 microsecond holds.

[0127] Fig. 12 is a flow chart showing an example of operations of the CIP generating part 115. Now, a content of operations of the CIP generating part 115 will be described with reference to Fig. 12.

[0128] After starting processing (Step 1), one frame of data are entered via the application 111 (Step 2), and frame data are partitioned into six DIF blocks from the beginning of the data (Step 3). Whether this is a DIF block at the beginning of the frame is judged (Step 4), and if this is at the beginning, a value calculated by the SYT calculating part 114 based on the frame number n (The number n is counted from 0.) is used as an SYT value (Step 5), but if this is at the beginning, the SYT value is used as FFFFH (Step 6). The SYT is set in a CIP header and added to the six DIF blocks, whereby a CIP is created (Step 7). After repeating the processing from Step 3 to Step 7 for one frame, the processing of each frame is terminated in accordance with a judgment at (Step 8), and a CIP for one frame is outputted to the IEEE1394 I/F 116 (Step 9). After repeating the processing from Step 2 to Step 9 for all data, the processing is terminated in accordance with a judgment at (Step 10) (Step 11).

[0129] Where a DV driver IEEE1394 I/F is entirely or partially realized by software for PC, because of a varying delay developed during acquisition of the time of day information on the bus indicated by the counter 117 on the bus, it has been heretofore difficult to calculate an accurate SYT value in a DV driver. In addition, a delay develops in the transmission start time as well, which gives rise to a situation that the receiving side can not receive a transmitted isochronous packet.

[0130] In contrast, since the DV driver 112 calculates an SYT as described above, it is possible to set, in all SYT, such values which enables the DV 118 to process frame data without fail, and hence, send data.

[0131] While the offset tso is 250 microseconds according to the above, the offset tso may be a value around this.

[0132] Further, although the foregoing has described that the transmission start time TS is set based on a value of the time of day information on the bus acquired by the transmission start time information holding part 113 from the counter 117 on the bus, depending on the accuracy of the transmission start time which can be designated for the IEEE1394 I/F 116, a fixed value may be used, in which case it is not necessary to acquire a value of the time of day information on the bus. For instance, if the mechanism is such a mechanism which can designate only the lower sixteen bits of the counter 117 on the bus, since it is not possible to delay the transmission start time sixteen cycles or more and time intervals remain almost unchanged in reality until the transmission start time to the DV 118, it is not necessary to confirm the time of day information on the bus.

[0133] Further, although the foregoing has described that frame data 708 are DV data which include video of the NTSC system, the frame data may be other stream data such as PAL video.

[0134] Further, although the foregoing has described that fractions resulting from calculation of SYT values are rounded off, fractions may be handled by a different fashion, such as counting fractions as one.

[0135] Further, the data transmission apparatus may be formed by hardware or software, or the both.

[0136] Further, the DV driver 112 according to this embodiment is an example of the data conversion means according to the present invention, and the IEEE1394 I/F 116 according to this embodiment is an example of the interface according to the present invention. In addition, the transmission start time TS according to this embodiment is an example of the transmission start time X for the first packet of the first frame according to the present invention, the offset tso according to this embodiment is an example of the initial value Z according to the present invention, and 1.001/30 second according to this embodiment is an example of the frame cycle Y according to the present invention.

[0137] According to this embodiment, since the transmission start time is constant and the processing start time is calculated using this value and the frame number, it is possible to transmit data in such a fashion which allows reception and processing without fail.

(Eighth Preferred Embodiment)

[0138] With respect to transmission of AV data on an IEEE1394 I/F, a transmission system which requires, from the viewpoint of copyright protection, to superimpose copyright information upon a packet header of an isochronous packet was proposed in "5C Digital Transmission Content Protection White Paper" in July, 1998. Based on this copyright information, it is possible to determine whether received data are available for copying or not. Where data are encoded, key information which is necessary for a person at the receiving end to correctly decipher a code is transmitted separately through asynchronous transmission.

[0139] A method for transmitting a transport packet (TS packet) of MPEG2 on IEEE 1394 will be described with reference to Fig. 18. As shown in Fig. 18 (a). a TS packet itself has a length of 188 bytes and a sync byte having a fixed pattern (47H) of one byte is added to MPEG2 data of 187 bytes. A source packet header is added to this TS packet, whereby a data packet is formed. A time stamp of 25 bits is set in the source packet header. The remaining seven bits are reserved. The time stamp is time of day information which indicates the timing of transmitting the TS packet. One data packet is divided or a plurality of data packets are combined with each other based on a transmission rate regarding data to be transmitted, thereby creating data blocks. A data packet is divided when the transmission rate is low (The number of divisions can be selected from 1, 2 and 4.), whereas when the transmission rate is high, a plurality of data packets are combined with each other (The number of combinations can be selected from 1 through 5.).

[0140] As shown in Fig. 18 (b) , a CIP header is added to such a data packet as described above, whereby a CIP (Common Isochronous Packet) is created. Divide/combine information regarding how data packets and data blocks above are divided or combined can be set in the CIP header.

[0141] As shown in Fig. 18 (c) . a packet header, a header CRC and a data CRC are added to the CIP, whereby an isochronous packet is formed.

[0142] As shown in Fig. 18 (d) , the last four bits in the packet header are an area called an "SY area," and when copyright information is to be transmitted together with data, the copyright information is assigned to this. The details of the copyright information are EMI_flag of two bits and O/E_flag of one bit, and the remaining one bit is reserved. The EMI_flag is information which indicates whether copying is allowed, while the O/E_flag is information for normal deciphering which is used when transmission data are encoded.

[0143] On the other hand, DirectShow available from Microsoft Corporation which is used in a PC which uses Windows98 OS available from Microsoft Corporation provides an environment for handling stream data, such as video/speech data. on a PC, and an application must handle video/speech data of a recording apparatus or sending/receiving apparatus through the mechanism of DirectShow. However, DirectShow is characterized in that when two or more types of data are transmitted, the respective data transmissions work independently of each other, and therefore, it is necessary to ensure synchronization regarding which copyright information is set for which portion of video data and which portion of speech data.

[0144] Now, an eighth preferred embodiment of the present invention will be described with reference to Fig. 13.

[0145] Fig. 13 shows an example of a receiving apparatus, wherein denoted at 151a is an application, denoted at 152a is a copyright information multiplexing part, denoted at 154 is a copyright information extracting part, denoted at 155 is a data packet re-constructing part, denoted at 156 is a header separator part, and denoted at 157 is an IEEE1394 I/F. The receiving apparatus according to the eighth preferred embodiment is connected with a sending apparatus 158 via an IEEE1394 bus.

[0146] First, the IEEE1394 I/F 157 receives an isochronous packet sent from the sending apparatus 158 which is connected with the IEEE1394 I/F 157, and the packet is divided into a packet header and a CIP and outputted to the header separator part 156.

[0147] The header separator part 156 retrieves a data block and divide/combine information from the received CIP and outputs them to the data packet re-constructing part 155, while at the same time outputting the received header to the copyright information multiplexing part 154.

[0148] The data packet re-constructing part 155 restores a data packet from the data block based on the received divide/combine information, and outputs it to the copyright information multiplexing part 152a.

[0149] The copyright information multiplexing part 154 retrieves copyright information from an SY area of the received packet header, and outputs it to the copyright information multiplexing part 152a.

[0150] The copyright information multiplexing part 152a creates an input/output packet in which the copyright information is superimposed on a reserved area of the data packet as shown in Fig. 17 (a), and outputs the generated input/output packet to the application 151a.

[0151] As described above, according to this embodiment, since copyright information is superimposed on a reserved area of a packet header, it is possible to output both data and the copyright information to the application 151a at the same time.

[0152] Although this embodiment requires to superimpose copyright information in a reserved area, copyright information may be written in other place. For example, as shown in Fig. 17 (b), since a sync byte of an TS packet has a fixed value, the original value can be obtained easily even after written over with copyright information, and therefore, superimposing in this area is possible. Alternatively, as shown in Fig. 17 (c), a similar effect is obtainable when an area for writing in copyright information is added.

[0153] Further, although the foregoing has described that copyright information and MPEG2 data are handled simultaneously, the present invention is applicable to other information as well except for copyright information. However such information is set in an isochronous packet, the present invention remains applicable.

[0154] Further, although the foregoing has described that data to be transmitted are MPEG2 data . data to be transmitted may be other data. For instance, an IEEE1394 I/F can send and receive DV data as well by the same protocol as when handling MPEG2 data. Thus. the present invention is applicable even to DV data.

[0155] Further. although the foregoing has described that the interface is an IEEE1394 I/F. other interface may be used

[0156] Further, the receiving apparatus may be realized by software and/or hardware.

(Ninth Preferred Embodiment)

[0157] Next. an eighth preferred embodiment of the present invention will be described with reference to Fig. 14.

[0158] Fig. 14 shows an example of a receiving apparatus, wherein denoted at 151b is an application, denoted at 152b is a copyright information multiplexing part. denoted at 153 is a data packet re-constructing part (TS packet extracting part ???), denoted at 154 is a copyright information extracting part, denoted at 155 is a data packet re-constructing part, denoted at 156 is a header separator part. and denoted at 157 is an IEEE1394 I/F. The receiving apparatus according to the ninth preferred embodiment is connected with a sending apparatus 158 via an IEEE1394 bus. In Fig. 14. the copyright information extracting part 154, the data packet re-constructing part 155, the header separator part 156 and the IEEE1394 I/F 157 are the same as those described in relation to the eighth preferred embodiment.

[0159] The IEEE1394 I/F 157, first. receives an isochronous packet sent from the sending apparatus 158 which is connected with the IEEE1394 I/F 157, and the packet is divided into a packet header and a CIP and outputted to the header separator part 156. which is similar to that in the eighth preferred embodiment.

[0160] Another similarity to the eighth preferred embodiment is that the header separator part 156 retrieves a data block and divide/combine information from the received CIP and outputs them to the data packet reconstructing part 155, while at the same time outputting the received header to the copyright information multiplexing part 154.

[0161] The data packet re-constructing part 155 restores a data packet from the data block based on the received divide/combine information and outputs it to the TS packet extracting part 153. and the TS packet extracting part 153 extracts a TS packet portion from the received data packet and outputs it to the copyright information multiplexing part 152b.

[0162] The copyright information multiplexing part 154 retrieves copyright information from an SY area of the received packet header, and outputs it to the copyright information multiplexing part 152b.

[0163] The copyright information multiplexing part 152b, as shown in Fig. 17 (d), replaces a sync byte of a TS packet with copyright information and accordingly creates an input/output packet. and outputs it to the application 151b.

[0164] As described above, according to this embodiment, since a sync byte of a TS packet is replaced with copyright information, it is possible to output both data and the copyright information to the application 151b at the same time.

[0165] Although this embodiment requires to replace a sync byte of a TS packet with the copyright information 114, copyright information may be written in other place. For example, as shown in Fig. 17 (e), a similar effect is obtainable when an area for writing in copyright information is added.

[0166] Further, although the foregoing has described that copyright information and MPEG2 data are handled simultaneously, the present invention is applicable to other information as well except for copyright information. However such information is set in an isochronous packet, the present invention remains applicable.

[0167] Further. although the foregoing has described that data to be transmitted are MPEG2 data . data to be transmitted may be other data. For instance, an IEEE1394 I/F can send and receive DV data as well by the same protocol as when handling MPEG2 data. Thus, the present invention is applicable even to DV data.

[0168] Further, although the foregoing has described that the interface is an IEEE1394 I/F. other interface may be used.

[0169] Further, the sending apparatus may be realized by software and/or hardware.

(Tenth Preferred Embodiment)

[0170] Next, a tenth preferred embodiment of the present invention will be described with reference to Fig. 15.

[0171] Fig. 15 shows an example of a sending apparatus, wherein denoted at 161a is an application, denoted at 162a is a copyright information multiplexing part, denoted at 164 is a data block generating part. denoted at 165 is a CIP generating part, and denoted at 166 is an IEEE1394 I/F. The sending apparatus according to the tenth preferred embodiment is connected with a receiving apparatus 167 via an IEEE1394 bus.

[0172] The application 161a outputs an input/output packet as that shown in Fig. 17 (a), for instance, to the copyright information multiplexing part 162a. The copyright information multiplexing part 162a extracts copyright information from the received input/output packet and outputs it to the CIP generating part 165, and at the same time, converts the input/output packet into a data packet and outputs it to the data block generating part 164. A portion which used to hold the copyright information may be only replaced with the pattern of the reserved, whereby conversion into a data packet is realized.

[0173] The data block generating part 164 determines to divide or combine the received data packet, converts the data packet into a data block, sets divide/combine information and outputs the data block and the divide/combine information to the CIP generating part 165.

[0174] The CIP generating part 165 adds a CIP header in which the received divide/combine information is set to the received data block, accordingly creates a CIP, and outputs it to the IEEE1394 I/F 166. The copyright information as well is outputted to the IEEE1394 I/F 166.

[0175] The IEEE1394 I/F 166 creates an isochronous packet based on the received CIP and copyright information and sends it to the receiving apparatus 167.

[0176] As described above, according to this embodiment, since an input/output packet superimposed on copyright information is used in a reserved area of a source packet header of a data packet and outputted via the application 161b, it is possible to handle both data and the copyright information in synchronization to each other.

[0177] Although this embodiment requires to superimpose copyright information in a reserved area, copyright information, may be written in other place. For example, as shown in Fig. 17(b), since a sync byte of an TS packet has a fixed value, the original value can be obtained easily even after written over with copyright information, and therefore, superimposing in this area is possible. Alternatively, as shown in Fig. 17(c), a similar effect is obtainable when an area for writing in copyright information is added.

[0178] Further, although the foregoing has described that copyright information and MPEG2 data are handled simultaneously, the present invention is applicable to other information as well except for copyright information. However such information is set in an isochronous packet, the present invention remains applicable.

[0179] Further, although the foregoing has described that data to be transmitted are MPEG2 data, data to be transmitted may be other data. For instance, an IEEE1394 I/F can send and receive DV data as well by the same protocol as when handling MPEG2 data. Thus, the present invention is applicable even to DV data.

[0180] Further, although the foregoing has described that the interface is an IEEE1394 I/F, other interface may be used.

[0181] Further, the sending apparatus may be realized by software and/or hardware.

(Eleventh Preferred Embodiment)

[0182] Next, an eleventh preferred embodiment of the present invention will be described with reference to Fig. 16.

[0183] Fig. 16 shows an example of a sending apparatus, wherein denoted at 161b is an application, denoted at 162b is a copyright information extracting part, denoted at 163 is a data packet generating part, denoted at 164 is a data block generating part, denoted at 165 is a CIP generating part, and denoted at 166 is an IEEE1394 I/F. The sending apparatus according to the eleventh preferred embodiment is connected with a receiving apparatus 167 via an IEEE1394 bus. In Fig. 16, the data block generating part 164, the CIP generating part 165, the IEEE1394 bus 166, and the receiving apparatus 167 are the same as those described in relation to the tenth preferred embodiment.

[0184] The application 161b outputs an input/output packet as that shown in Fig. 17 (d), for instance, to the copyright information extracting part 162b. The copyright information extracting part 162b extracts copyright information from the received input/output packet and outputs it to the CIP generating part 165, and at the same time, converts the input/output packet into a TS packet and outputs it to the data packet generating part 163. A portion which used to hold the copyright information may be only replaced with the pattern of a sync byte, whereby conversion into a TS packet is realized.

[0185] The data packet generating part 163 creates a time stamp, converts the TS packet into a data packet, and outputs it to the data block generating part 164. The time stamp is created based on time information which is in a TS packet (known as "PCR") so as to be restored on the receiving side. The PCR is prepared in such a manner that an MPEG2 decoder can restore video and speech at the correct timing.

[0186] The data block generating part 164 determines to divide or combine the received packet, converts the data packet into a data block, sets divide/combine information and outputs the data block and the divide/combine information to the CIP generating part 165, which is similar to that in the tenth preferred embodiment.

[0187] Another similarity to the tenth preferred embodiment is that the CIP generating part 165 outputs to the IEEE1394 I/F 166 copyright information and a CIP which is created by adding a CIP header in which the received divide/combine information is set to the received data block.

[0188] Yet another similarity to the tenth preferred embodiment is that the IEEE1394 I/F 166 creates an isochronous packet based on the received CIP and copyright information and sends it to the receiving apparatus 167.

[0189] Although this embodiment requires to replace a sync byte of a TS packet with copyright information, copyright information may be written in other place. For example, as shown in Fig. 17 (e), a similar effect is obtainable when an area for writing in copyright information is added.

[0190] Further, although the foregoing has described that the data packet generating part 163 creates a time stamp based on PCR, as far as the application 161b is capable of outputting at the correct timing owing to software or hardware, it is also possible to create a time stamp from a time at which a TS packet 204 is received by the data packet generating part 163.

[0191] Further, although the foregoing has described that copyright information and MPEG2 data are handled simultaneously, the present invention is applicable to other information as well except for copyright information. However such information is set in an isochronous packet, the present invention remains applicable.

[0192] Further, although the foregoing has described that data to be transmitted are MPEG2 data, data to be transmitted may be other data. For instance, an IEEE1394 I/F can send and receive DV data as well by the same protocol as when handling MPEG2 data. Thus, the present invention is applicable even to DV data.

[0193] Further, although the foregoing has described that the interface is an IEEE1394 I/F other interface may be used.

[0194] Further, the sending apparatus may be realized by software and/or hardware.

(Twelfth Preferred Embodiment)

[0195] The present invention is realized by a program for a computer, a microcomputer or the like, and if recorded in a recording medium such as a floppy disk and transported, the present invention is implemented easily in other independent system. Fig. 19 is a drawing for describing implementation using a floppy disk.

[0196] Fig. 19 (a) shows an example of a physical format of a floppy disk which is a recording medium main part. Tracks are formed in a concentric circle arrangement inwardly from outside and divided into sixteen sectors in angle directions. A program is recorded in accordance with areas which are allocated in this fashion.

[0197] Fig. 19 (b) is a drawing for describing a case which houses this floppy disk. From the left-hand side are the floppy disk case as viewed from the front, the floppy disk case in cross section, and the floppy disk. With the floppy disk thus housed in the case, the disk is protected against dust and external impact and transported safely.

[0198] Fig. 19(c) is a drawing for describing recording and reproduction of a program on a floppy disk. As a floppy disk drive is connected with a computer system in a manner as that shown in this drawing, it is possible to record and reproduce a program on a disk. A disk is loaded in and unloaded from the floppy disk drive through an insertion slot. With the program recorded on the disk from the computer system via the floppy disk drive, the program for realizing the present invention is saved. In order to implement the present invention, the floppy disk drive reads the program from the disk, transfers to the computer system and creates an environment for realizing the present invention.

[0199] Although the foregoing has described that a floppy disk is used as a recording medium in this embodiment, an optical disk may be used for a similar end. In addition, a recording medium is not limited to this but may be any other on which the program can be recorded, such as an IC card, a ROM and a cassette, for similar implementation.

[0200] As here in referred to, "data" include a data structure, a data format, a data type, etc.

[0201] "Mediums" include recording mediums such as a ROM, transmission mediums such as the Internet, and transmission mediums such as light, electric waves and acoustic waves.

[0202] "Carried mediums" include a recording medium in which a program and/or data are recorded, transmission mediums for transmitting a program and/or data, and the like.

[0203] "Can be processed on a computer" covers, in the case of a recording medium such as a ROM, that the medium can be read on a computer, and in the case of a transmission medium, that a computer can handle the program and/or data to be transmitted as a result of transmission.

[0204] The present invention may be directed to a program recording medium in which a program and/or data for making a computer execute all or some functions of all or some means of the structure elements described in relation to the preferred embodiments above is recorded, which can be read on a computer so that the read-out program and/or data execute the functions while working in cooperation with the computer.

[0205] Further, an aggregation of information includes software such as a program and/or data, for example.

AVAILABILITY FOR INDUSTRIAL USE

[0206] As is clear from the above, the present invention solves a problem with the conventional techniques that a DV, sending and receiving stream data, can not respond to an asynchronous random access described earlier and a mechanism for dealing with such an access is necessary.

[0207] In addition, the present invention solves a problem that despite the recent ongoing fusion of PCs and digital video/audio equipment such as DVs, since PCs have never initially considered handling synchronous data, the current mechanisms can not handle real-time data without a break.

[0208] Moreover, the present invention meets a challenge that a new mechanism to handle AV data in accordance with copyright information is necessary to send and receive AV data between PCs and other AV equipment.

Claims

1. A data processing system comprising an operating system which contains a file system driver and a device driver,

   characterized in comprising:

   a real-time interface driver which processes data of a device which comprises an interface which is required to ensure real-time processing during recording and/or reproducing;

a non real-time interface driver which processes data of a device which comprises an interface which is not required to ensure real-time processing during recording and/or reproducing;

an IFS manager which allocates instructions for predetermined processing regarding a file system to each one of control means which are to handle the instructions;

first control means which, using first file management information with respect to data which are in a first medium which is housed in the device which comprises an interface which is required to ensure said real-time processing, converts an instruction for said predetermined processing inputted through said IFS manager into a data processing instruction on said first medium and sends to said real-time interface driver; and

second control means which, using second file management information with respect to data which are in a second medium which is housed in the device which comprises an interface which is not required to ensure said real-time processing, executes processing for accessing to a file on said second medium in accordance with the instruction for said predetermined processing inputted through said IFS manager and sends the instruction for said predetermined processing to said non real-time interface driver via an I/O sub system.

2. The data processing system of Claim 1, **characterized in that** said first control means comprises:

a file system driver for first medium which judges whether it is necessary to access a file on said first medium based on the instruction for said predetermined processing inputted through said IFS manager and said first file management information; and

file system processing support application means which creates a data processing instruction on said first medium when an access is necessary in accordance with the judgment made by said file system driver for first medium,

and **in that** said file system processing support application means drives said real-time interface driver.

3. The data processing system of Claim 2, **characterized in that** said file system processing support application means, using information from an IFS monitor which monitors a state of an instruction for predetermined processing which is handled by said IFS manager, creates a data processing instruction on said first medium.

4. The data processing system of Claim 2, **characterized in that** said file system driver for first medium comprises:

a file management information holding part which stores said file management information;

a identifying part which receives an instruction from said IFS manager and identifies said instruction;

a file entry extracting part which extracts a file entry corresponding to a file name from said file management information holding part when said instruction is OPEN, registers in an access file entry holding part, and sends said instruction to said file system processing support application means;

a second file information extracting part which retrieves information regarding a file which is designated by said access file entry holding part based on a file ID received together with said instruction when said instruction is READ, WRITE or CLOSE, adds the information to said instruction, and sends to said file system processing support application means, and

a first file information extracting part which retrieves information regarding a file which is designated by said access file entry holding part based on said file ID received together with said instruction when said instruction is an instruction except for OPEN, READ, WRITE and CLOSE, and sends to said IFS manager through said identifying part.

5. The data processing system of Claim 3, **characterized in that** said file system processing support application means comprises:

a support processing judging part which determines a content of processing based on information from said file system driver for first medium and information from said IFS monitor and instructs; and

a copying part which executes copying processing in accordance with the judgment made by said support processing judging part.

6. The data processing system of Claim 3, **characterized in that** said first file management information is recorded on said first medium,

and said file system processing support application means comprises a file management information updating/acquiring part which is for reading said first file management information from said first medium or for writing on said first medium in accordance with the judgment made by said support processing judging part.

7. The data processing system of Claim 5 or 6, **characterized in that** said file system processing support application means further comprises a random access processing part which

receives an OPEN, a CLOSE, a READ and a WRITE instructions other than instructions which are related to copying processing,
when said instruction is an OPEN instruction or a CLOSE instruction, receives said instruction and information regarding a file which is designated,
and when said instruction is a READ instruction or a WRITE instruction, receives and processes information regarding which portion of said file is demanded by said instruction, in addition to said instruction and said information regarding a file which is designated.

8. A data processing system comprising an operating system which contains a file system driver and a device driver,
characterized in comprising:

display application means which displays files which can be handled in a file system, selects a desired file from the displayed files, and starts processing which is associated with said selected file;
a first processing part which executes processing on a first medium which stores data of a device which comprises an interface which is required to ensure real-time processing during recording and/or reproducing;
a second processing part which executes processing on a second medium which stores data of a device which comprises an interface which is not required to ensure real-time processing during recording and/or reproducing;
device identifying means which identifies a device in which said selected file exists; and
a file starter part which drives said first or said second processing part in accordance with the identification result made by said device identifying means.

9. A data processing system comprising an operating system which contains a file system driver and a device driver,
characterized in comprising:

a real-time interface driver which processes data of a device which comprises an interface which is required to ensure real-time processing during recording and/or reproducing;
a non real-time interface driver which processes data of a device which comprises an interface which is not required to ensure real-time processing during recording and/or reproducing;
an IFS manager which allocates instructions for predetermined processing regarding a file system to each one of control means which are to handle the instructions;
first control means which, using first file management information with respect to data of the device which comprises an interface which is required to ensure said real-time processing which are stored in a first medium, converts an instruction for said predetermined processing inputted through said IFS manager into a data processing instruction on said first medium and sends to said real-time interface driver; and
second control means which, using second file management information with respect to data of the device which comprises an interface which is not required to ensure said real-time processing which are stored in a second medium, executes processing for accessing to a file on said second medium in accordance with the instruction for said predetermined processing inputted through said IFS manager and sends the instruction for said predetermined processing to said non real-time interface driver via an I/O sub system;
display application means which displays files which can be handled in a file system, selects a desired file from the displayed files, and starts processing which is associated with said selected file;
a first processing part which executes processing on a first medium which stores data of a device which comprises an interface which is required to ensure real-time processing during recording and/or reproducing;
a second processing part which executes processing on a second medium which stores data of a device which comprises an interface which is not required to ensure real-time processing during recording and/or reproducing;
drive identifying means which identifies a drive in which said selected file exists; and
a file starter part which drives said first or said second processing part in accordance with the identification result made by said device identifying means.

10. The data processing system of any one of Claims 1 through 9, **characterized in that** said real-time data are video data.

11. A medium which can be processed on a computer

and which carries a program and/or data for making a computer execute all or some functions of all or some means according to any one of Claims 1 through 10.

12. An aggregation of information which is a program and/or data for making a computer execute all or some functions of all or some means of the invention according to any one of Claims 1 through 10.

13. A data transmission apparatus, comprising:

time intervals generating means which creates predetermined time of day information;
outputting means which outputs a series of data frames as stream data;
sending means which divides said series of data frames forming said stream into packet data and sends; and
transmission managing means which manages said outputting means and said sending means,

characterized in that said transmission managing means enters said stream data to said sending means based on said predetermined time of day information.

14. The data transmission apparatus of Claim 13, **characterized in that** said time of day information is set up to correspond to necessary timing which is needed by an apparatus which receives packet data of said series of data frames.

15. The data transmission apparatus of Claim 14, **characterized in that** said necessary timing is based on a frame frequency of said series of data frames.

16. The data transmission apparatus of any one of Claims 13 through 15, **characterized in that** said transmission managing means provides said time intervals generating means with a correction instruction which is for correcting said predetermined time of day information in accordance with a condition of burden upon said sending means,

and said time intervals generating means creates said predetermined time of day information without receiving the correction instruction or considering the correction instruction.

17. The data transmission apparatus of any one of Claims 13 through 15, **characterized in that** said transmission managing means does not provide said time intervals generating means with a correction instruction which is for correcting said predetermined time of day information in accordance with a condition of burden upon said sending means.

18. The data transmission apparatus of any one of Claims 13 through 17, **characterized in that** said predetermined time of day information time is expressed as groups of the start time and the end time of said data frames.

19. The data transmission apparatus of any one of Claims 13 through 18, **characterized in that** said stream data are data for a digital VCR for consumer use.

20. The data transmission apparatus of any one of Claims 13 through 19, **characterized in that** said sending means sends said stream data to a digital VCR for consumer use.

21. The data transmission apparatus of any one of Claims 13 through 20, **characterized in that** said outputting means reproduces data for a digital VCR for consumer use.

22. A data transmission apparatus, comprising:

data conversion means which divides inputted stream data and adds header information to each piece to thereby obtain packets, and inserts packet processing start time information which is for the data receiving side into the header information of at least first packet of each frame of said stream data and thereafter outputs; and
an interface which, using a clock, outputs to a bus the packets processed by said data conversion means, at the transmission start time which corresponds to said packet processing start time information,

characterized in that the packet processing start time information of said packets is expressed by:

$$T1 = X + Z + Y(N - 1)$$

(where X >= 0, Z >= 0) assuming that X denotes the transmission start time for the first packet of the first frame, N denotes a frame number, Y denotes a frame period, Z denotes an initial value, and T1 denotes the processing start time of said packets.

23. The data transmission apparatus of Claim 22, **characterized in that** said bus is an IEEE1394 bus, and said interface is an IEEE1394 interface.

24. The data transmission apparatus of Claim 22 or 23, **characterized in that** said stream data are data for a digital VCR for consumer use, said Z is a value around 250 microseconds, and said Y is a value

which is based on a frame frequency of said stream data.

25. A medium which can be processed on a computer and which carries a program and/or data for making a computer execute all or some functions of all or some means of the data transmission apparatus of any one of Claims 13 through 24.

26. An aggregation of information which is a program and/or data for making a computer execute all or some functions of all or some means of the invention according to any one of Claims 13 through 24.

27. A receiving apparatus, comprising:

an interface which receives a transmission packet which contains a transmission path header in which additional information is described and data blocks;
transmission path header separator means which separates said transmission packet into said transmission path header and said data blocks;
additional information extracting means which extracts said additional information from said transmission path header;
data packet generating means which generates from said data blocks a data packet which is obtained by combining one or more of said data blocks; and
additional information inserting means which adds or inserts said additional information to said data packet and outputs as an output packet of one type of data format which can be processed by an application simultaneously,

characterized in that said application processes the output packet outputted from said additional information inserting means.

28. The receiving apparatus of Claim 27, **characterized in that** said data packet is formed by a source packet header and a transport packet in MPEG2 data.

29. The receiving apparatus of Claim 27 or 28, **characterized in that** said additional information inserting means adds additional information to the beginning or end of said data packet.

30. The receiving apparatus of Claim 28, **characterized in that** said additional information inserting means inserts said additional information to said source packet header.

31. A receiving apparatus, comprising:

an interface which receives a transmission packet which contains a transmission path header in which additional information is described and data blocks;
transmission path header separator means which separates said transmission packet into said transmission path header and said data blocks;
additional information extracting means which extracts said additional information from said transmission path header;
data packet generatingmeans which generates from said data blocks a data packet which is obtained by combining one or more of said data blocks;
source packet header separator means which separates said data packet into a source packet header and a source packet; and
additional information inserting means which adds or inserts said additional information to said source packet or replaces said additional information, and outputs as an output packet of one type of data format which can be be processed by an application simultaneously,

characterized in that said application processes the output packet outputted from said additional information inserting means.

32. The receiving apparatus of Claim 31, **characterized in that** said source packet is a transport packet in MPEG2 data.

33. The receiving apparatus of Claim 32, **characterized in that** said additional information inserting means adds said additional information to the beginning or end of said transport packet.

34. The receiving apparatus of Claim 32, **characterized in that** said additional information inserting means replaces a sync byte of said transport packet with said additional information.

35. The receiving apparatus of any one of Claims 27 through 34, **characterized in that** said interface is an IEEE1394 interface, and said transport packet is an isochronous packet.

36. The receiving apparatus of any one of Claims 27 through 35, **characterized in that** said additional information is copyright information.

37. The receiving apparatus of any one of Claims 27 through 36, **characterized in that** said additional information is described in an SY area of an isochronous packet.

38. A data sending apparatus, comprising:

additional information separator means which, upon receipt of an input packet of one type of data format which can be processed by an application simultaneously, retrieves said additional information which is added or inserted to a data packet which forms said input packet;
data block generating means which generates one or more data blocks from said data packet;
transmission packet generating means which executes predetermined processing on said data blocks, inserts said additional information at a predetermined location in said data blocks, and accordingly generates said transmission packet; and
an interface which sends said transmission packet.

39. The sending apparatus of Claim 38, **characterized in that** said data packet is formed by a source packet header and a transport packet in MPEG2 data.

40. The sending apparatus of Claim 38 or 39, **characterized in that** in said input packet, said additional information is added to the beginning or end of said data packet.

41. The sending apparatus of Claim 39, **characterized in that** said input packet inserts said additional information to said source packet header.

42. A sending apparatus, comprising:

additional information separator means which, upon receipt of an input packet of one type of data format which can be processed by an application simultaneously, constructs said input packet and retrieves said additional information from a transport packet which includes added, inserted or replaced additional information;
data packet generating means which combines a source packet header with said transport packet to thereby generate a data packet;
data block generating means which generates one or more data blocks from said data packet;
transmission packet generating means which executes predetermined processing on said data blocks, inserts said additional information at a predetermined location in said data blocks, and accordingly generates said transmission packet; and
an interface which outputs said transmission packet.

43. The sending apparatus of Claim 42, **characterized in that** said input packet is obtained by replacing a sync byte of said transport packet with additional information.

44. The sending apparatus of Claim 42, **characterized in that** said input packet is obtained by adding said additional information to the beginning or end of said transport packet.

45. The sending apparatus of any one of Claims 38 through 44, **characterized in that** said interface is an IEEE1394 interface, and said transport packet is an isochronous packet.

46. The sending apparatus of any one of Claims 38 through 45, **characterized in that** said additional information is copyright information.

47. The sending apparatus of any one of Claims 38 through 46, **characterized in that** said additional information is described in an SY area of an isochronous packet.

48. medium which can be read on a computer and which holds a program for making a computer execute all or some of the respective means or the respective structural elements which form the receiving apparatus or the sending apparatus according to any one of Claims 27 through 47.

49. An aggregation of information which is a program and/or data for making a computer execute all or some functions of all or some means of the invention according to any one of Claims 27 through 47.

Fig. 1

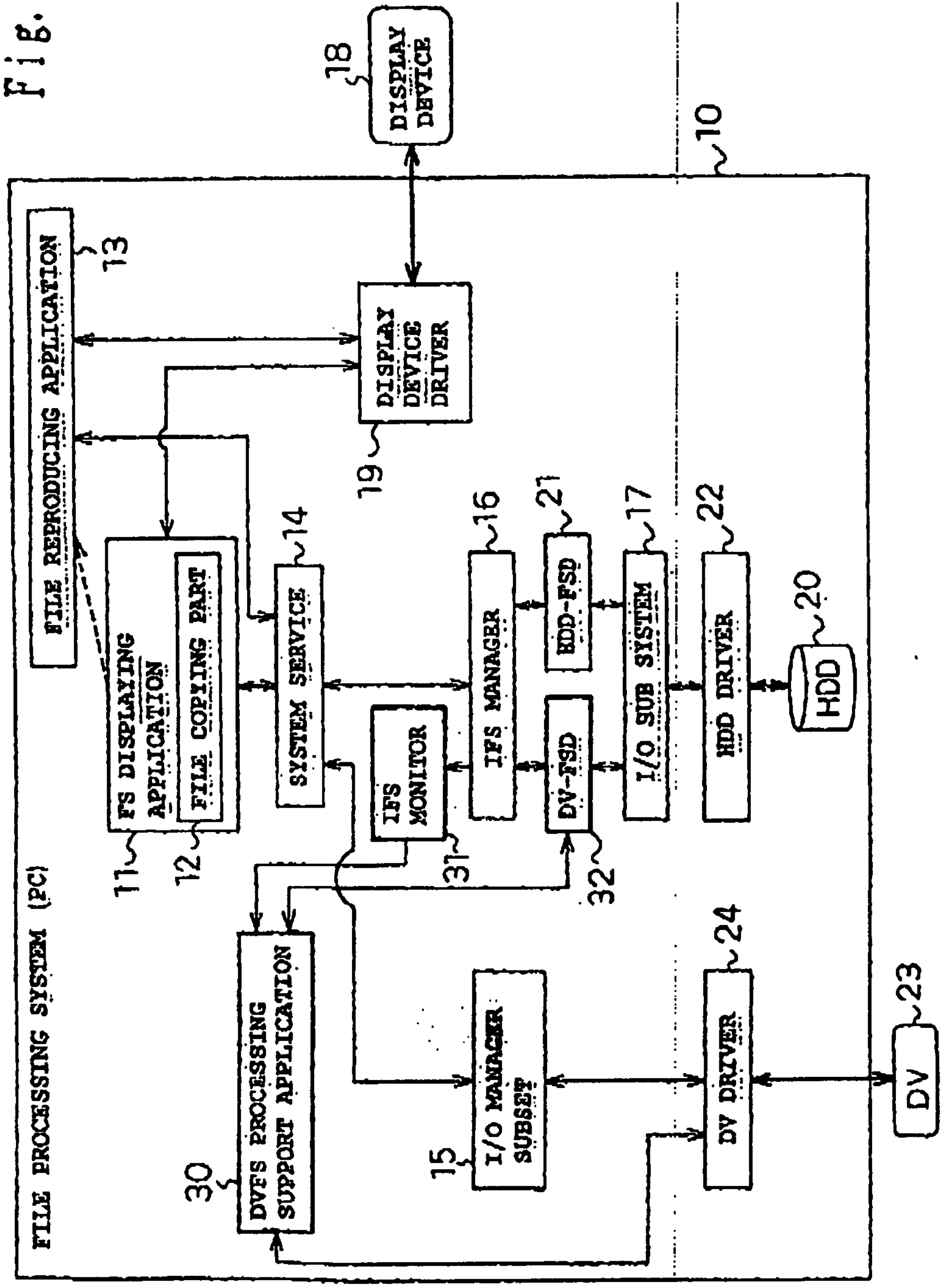

FILE PROCESSING SYSTEM (PC)
10
FILE REPRODUCING APPLICATION
13
FS DISPLAYING APPLICATION
11
FILE COPYING PART
12
SYSTEM SERVICE
14
DVFS PROCESSING SUPPORT APPLICATION
30
DISPLAY DEVICE DRIVER
19
DISPLAY DEVICE
18
IFS MONITOR
31
IFS MANAGER
16
I/O MANAGER SUBSET
15
HDD-FSD
21
DV-FSD
32
I/O SUB SYSTEM
17
HDD DRIVER
22
HDD
20
DV DRIVER
24
DV
23

Fig. 2

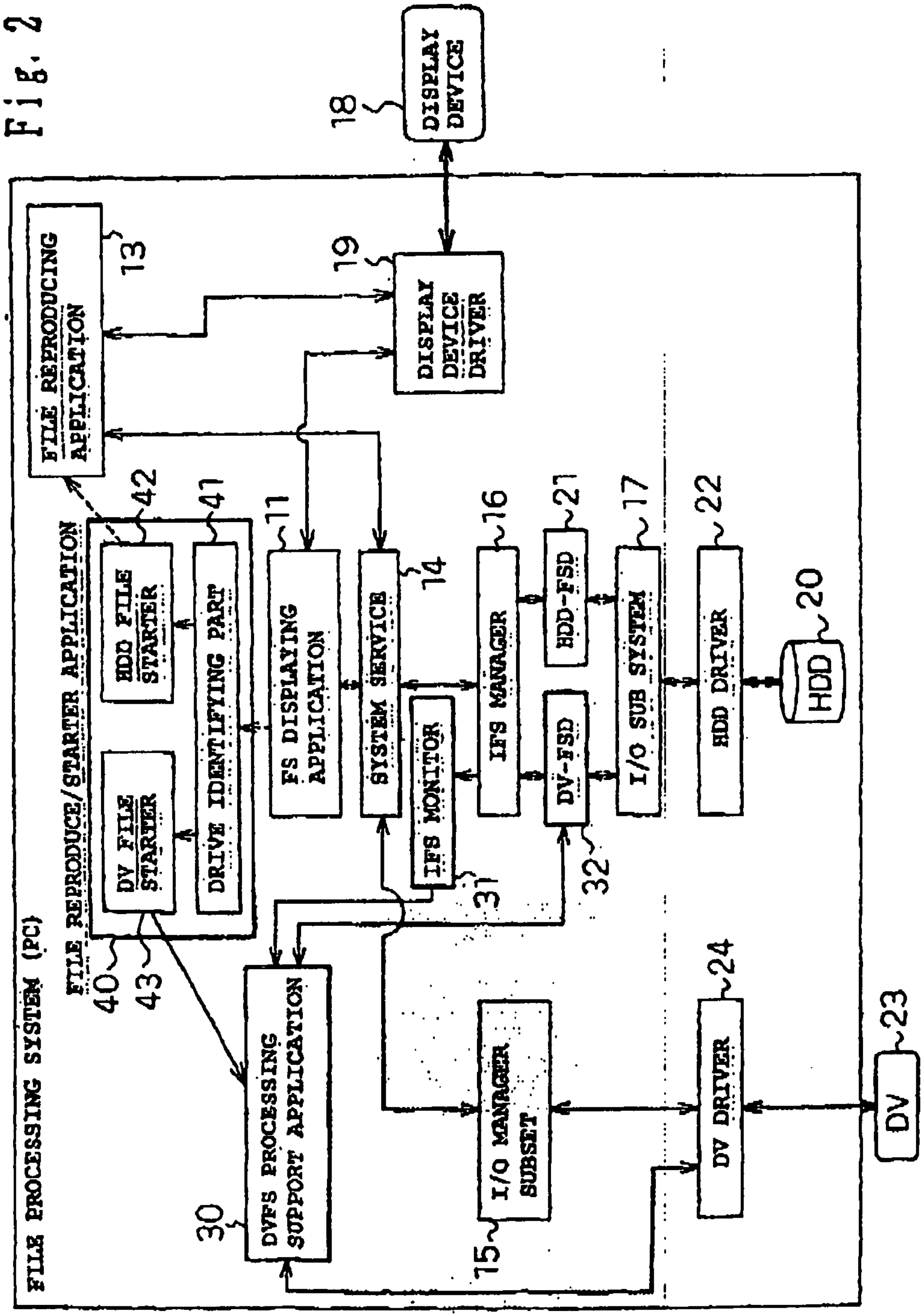

FILE PROCESSING SYSTEM (PC)
DISPLAY DEVICE
18
DISPLAY DEVICE DRIVER
19
FILE REPRODUCING APPLICATION
13
FILE REPRODUCE/STARTER APPLICATION
40
HDD FILE STARTER
42
DRIVE IDENTIFYING PART
41
DV FILE STARTER
43
FS DISPLAYING APPLICATION
11
SYSTEM SERVICE
14
IFS MONITOR
31
IFS MANAGER
16
HDD-FSD
21
DV-FSD
32
I/O SUB SYSTEM
17
HDD DRIVER
22
HDD
20
DVFS PROCESSING SUPPORT APPLICATION
30
I/O MANAGER SUBSET
15
DV DRIVER
24
DV
23

Fig. 3

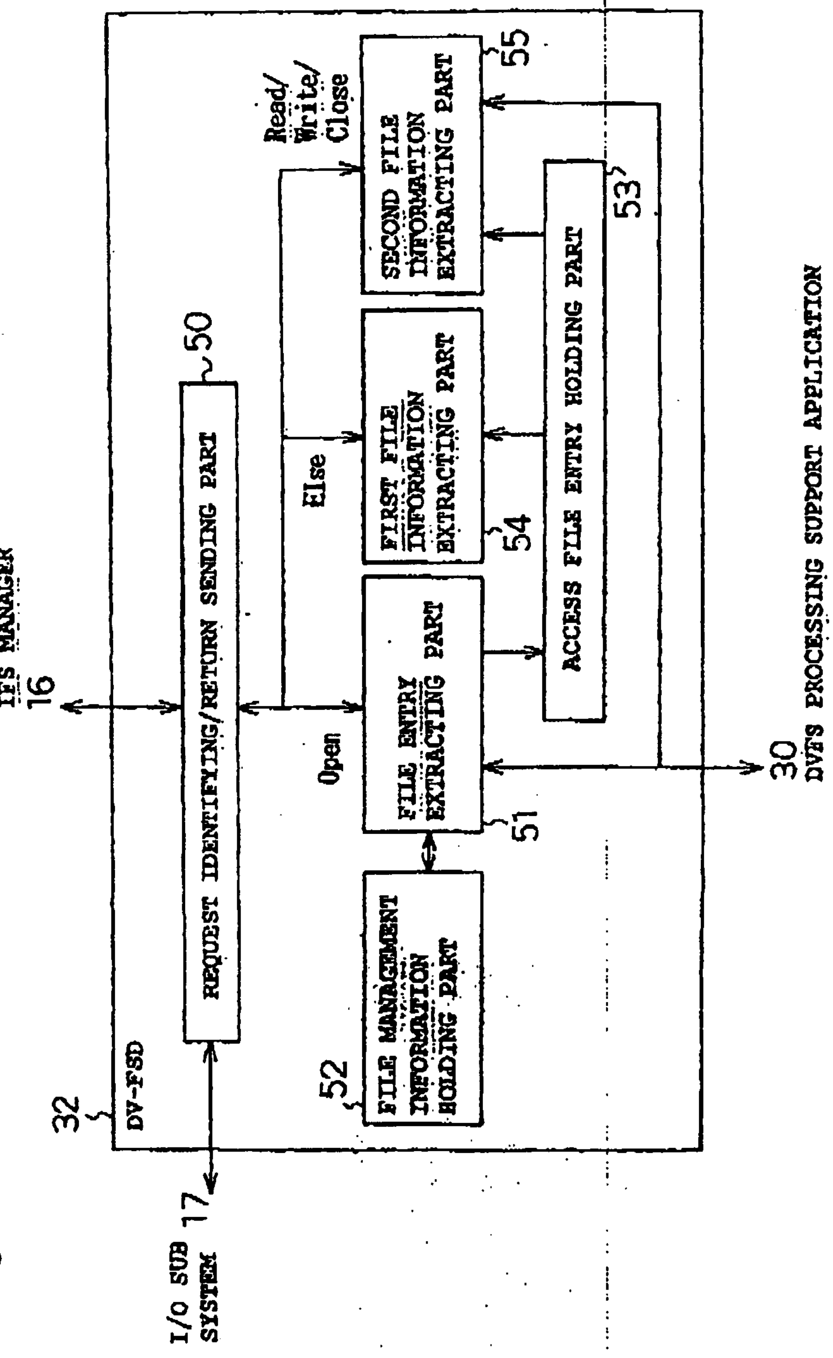
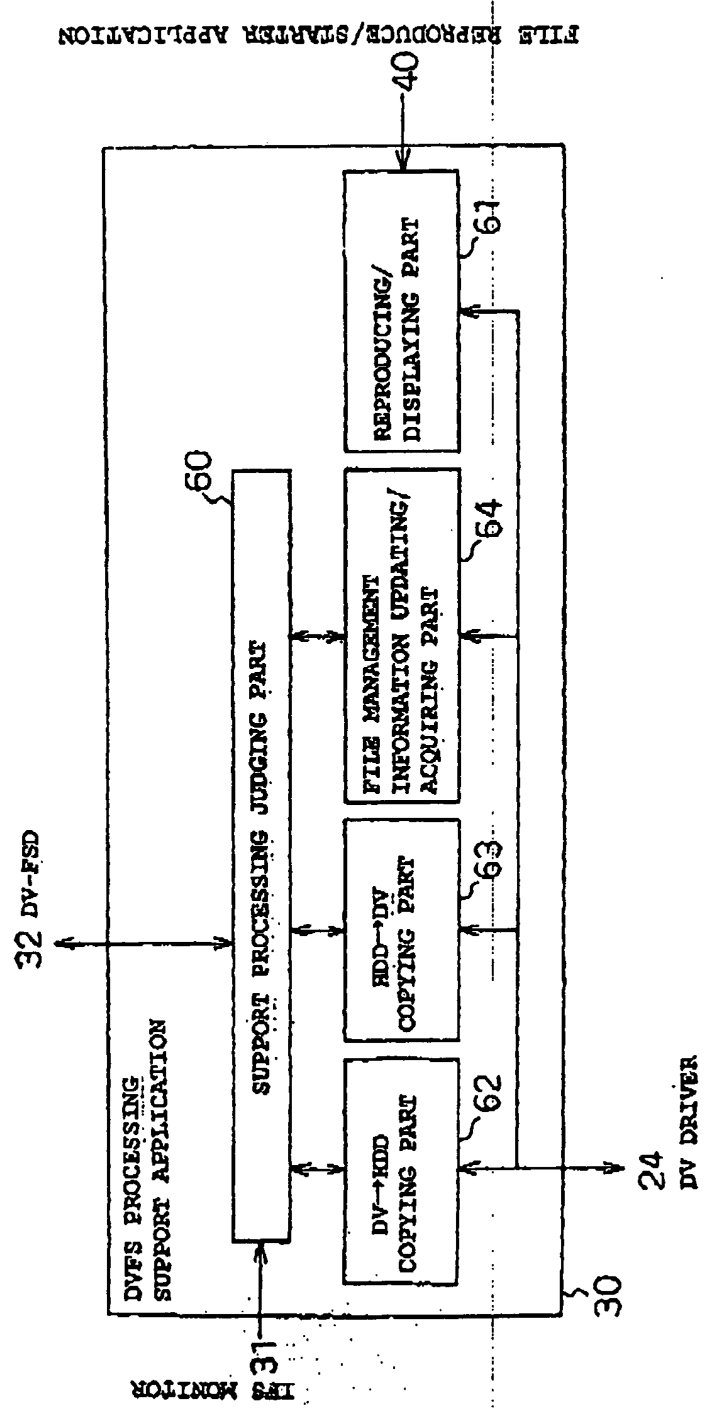

IFS MANAGER
16
32
DV-FSD
REQUEST IDENTIFYING/RETURN SENDING PART
50
17
I/O SUB SYSTEM
Open
Else
Read/Write/Close
FILE ENTRY EXTRACTING PART
51
FIRST FILE INFORMATION EXTRACTING PART
54
SECOND FILE INFORMATION EXTRACTING PART
55
FILE MANAGEMENT INFORMATION HOLDING PART
52
ACCESS FILE ENTRY HOLDING PART
53
30
DVFS PROCESSING SUPPORT APPLICATION

Fig. 4


FILE REPRODUCE/STARTER APPLICATION
40
DVFS PROCESSING SUPPORT APPLICATION
30
SUPPORT PROCESSING JUDGING PART
60
DV→HDD COPYING PART
62
HDD→DV COPYING PART
63
FILE MANAGEMENT INFORMATION UPDATING/ACQUIRING PART
64
REPRODUCING/DISPLAYING PART
61
32 DV-FSD
24 DV DRIVER
31 IFS MONITOR

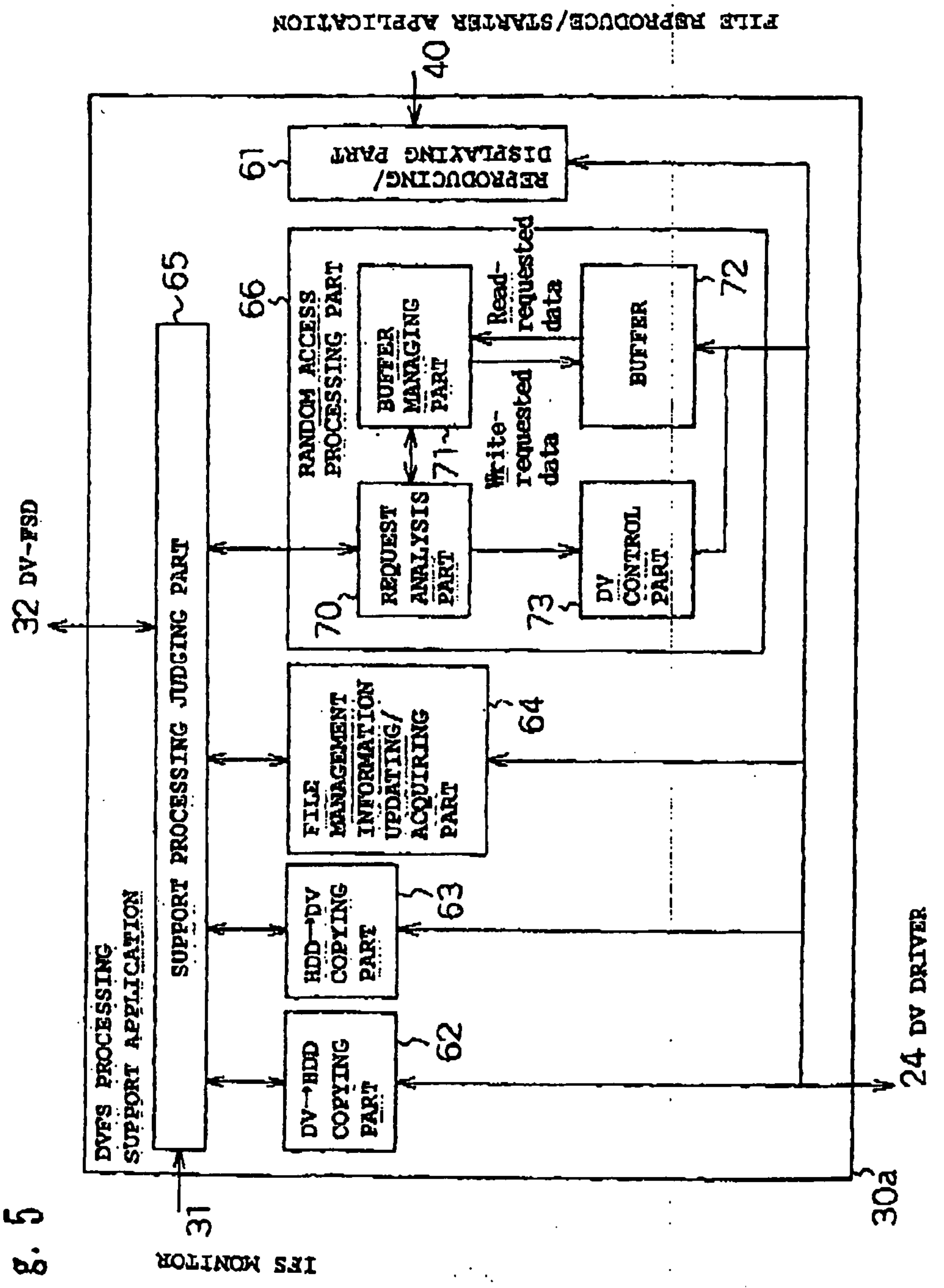

FILE REPRODUCE/STARTER APPLICATION
40
61
REPRODUCING/ DISPLAYING PART
66
RANDOM ACCESS PROCESSING PART
BUFFER MANAGING PART
71
Read-requested data
BUFFER
72
Write-requested data
70
REQUEST ANALYSIS PART
73
DV CONTROL PART
65
SUPPORT PROCESSING JUDGING PART
32 DV-FSD
64
FILE MANAGEMENT INFORMATION UPDATING/ ACQUIRING PART
63
HDD→DV COPYING PART
62
DV→HDD COPYING PART
DVFS PROCESSING SUPPORT APPLICATION
24 DV DRIVER
30a
31
IFS MONITOR

Fig. 5

Fig. 6

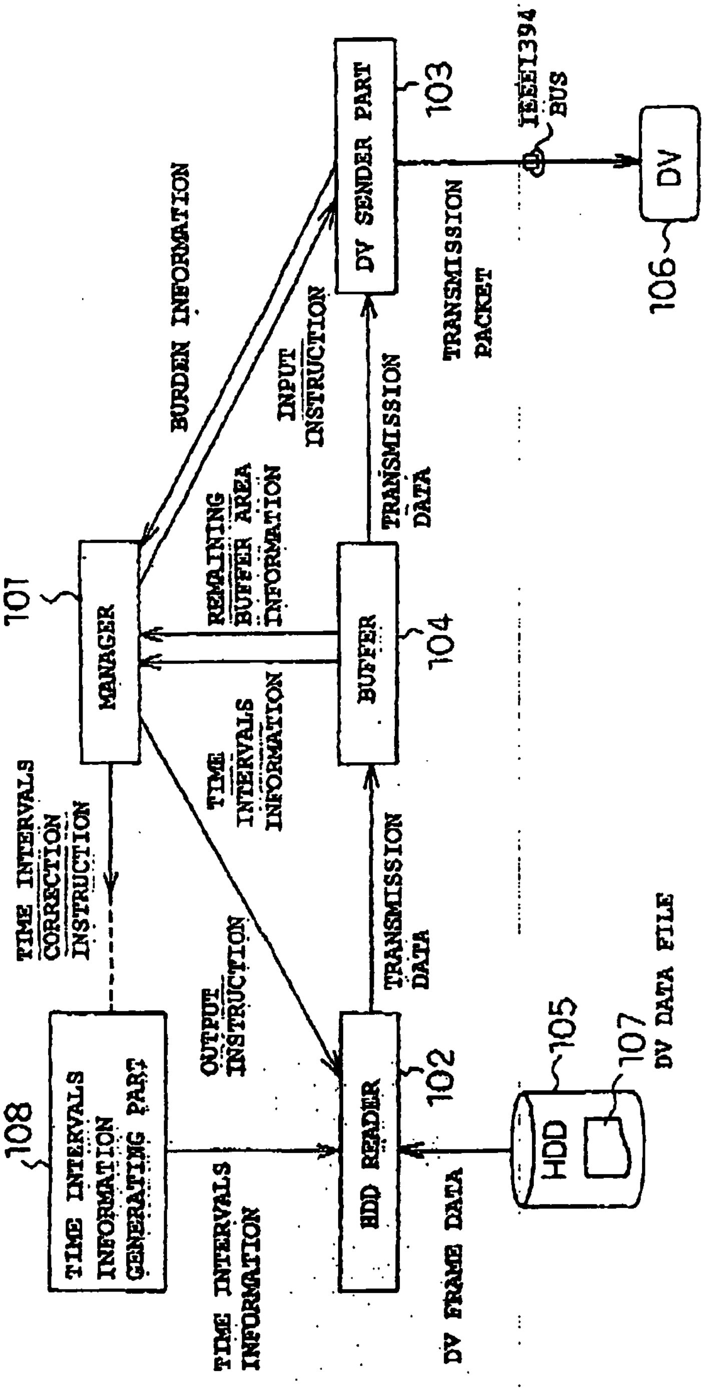

TIME INTERVALS INFORMATION GENERATING PART
108
HDD READER
102
HDD
105
107
DV DATA FILE
DV FRAME DATA
TIME INTERVALS INFORMATION
OUTPUT INSTRUCTION
TIME INTERVALS CORRECTION INSTRUCTION
TRANSMISSION DATA
MANAGER
101
TIME INTERVALS INFORMATION
REMAINING BUFFER AREA INFORMATION
BUFFER
104
TRANSMISSION DATA
BURDEN INFORMATION
INPUT INSTRUCTION
DV SENDER PART
103
TRANSMISSION PACKET
IEEE1394 BUS
DV
106

Fig. 7

FRAME NUMBER	TIME INTERVALS INFORMATION	
	START TIME	END TIME
0	0	1*1.001/30
1	1*1.001/30	2*1.001/30
2	2*1.001/30	3*1.001/30
3	3*1.001/30	4*1.001/30
:	:	:
n	n*1.001/30	(n+1)*1.001/30
:	:	:

Fig. 8

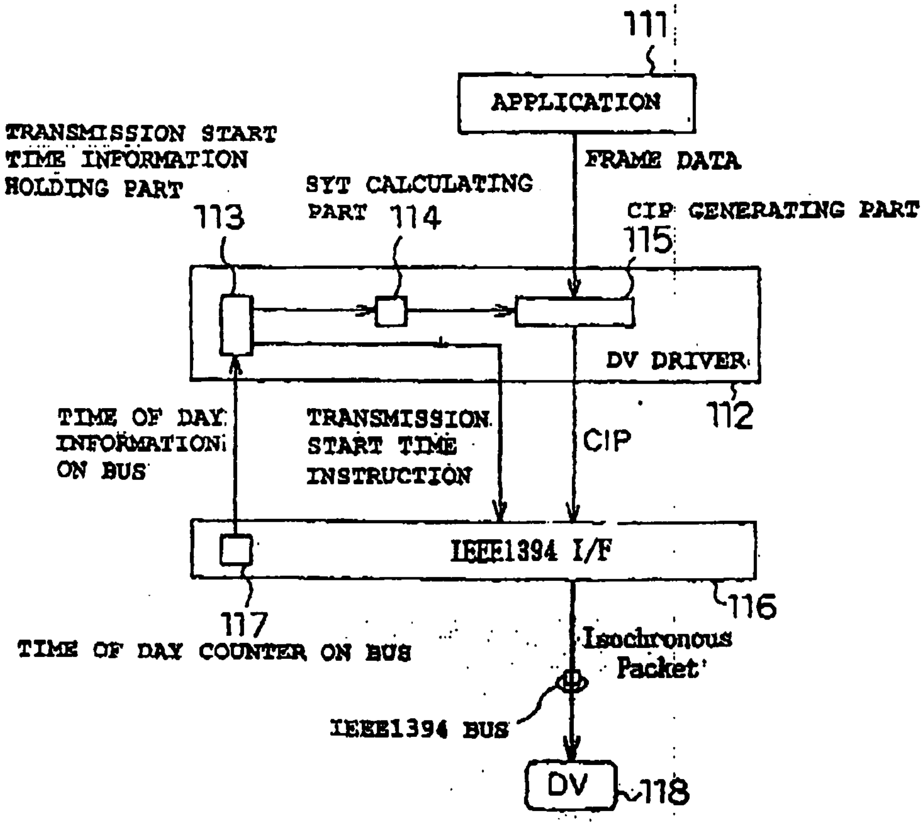

111
APPLICATION
FRAME DATA
TRANSMISSION START TIME INFORMATION HOLDING PART
113
SYT CALCULATING PART
114
CIP GENERATING PART
115
DV DRIVER
112
TIME OF DAY INFORMATION ON BUS
TRANSMISSION START TIME INSTRUCTION
CIP
IEEE1394 I/F
116
117
TIME OF DAY COUNTER ON BUS
Isochronous Packet
IEEE1394 BUS
DV
118

Fig. 9 (a)

Fig. 9 (b)

Fig. 9 (c)

Fig. 9 (d)

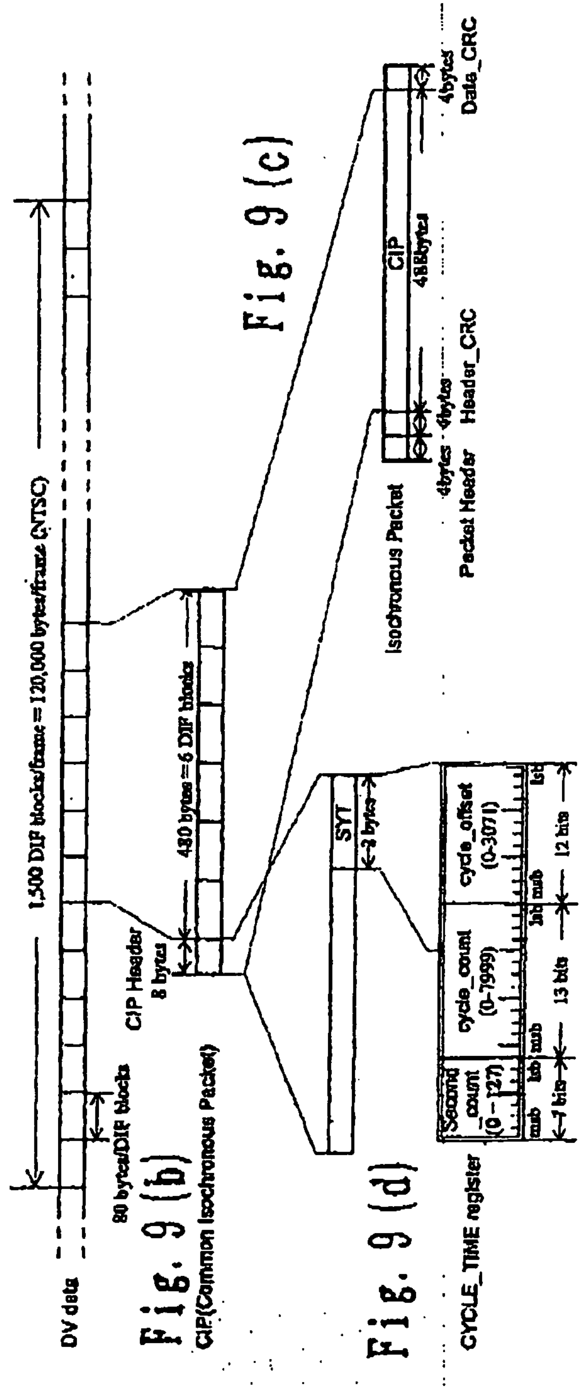

DV data
1,500 DIF blocks/frame = 120,000 bytes/frame (NTSC)
80 bytes/DIF blocks
CIP(Common Isochronous Packet)
CIP Header 8 bytes
480 bytes = 6 DIF blocks
Isochronous Packet
Packet Header
4bytes
Header_CRC
4bytes
CIP
488bytes
Data_CRC
4bytes
SYT
2 bytes
CYCLE_TIME register
Second count (0 - 127)
cycle_count (0-7999)
cycle_offset (0-3071)
msb
lsb
7 bits
13 bits
12 bits

Fig. 10

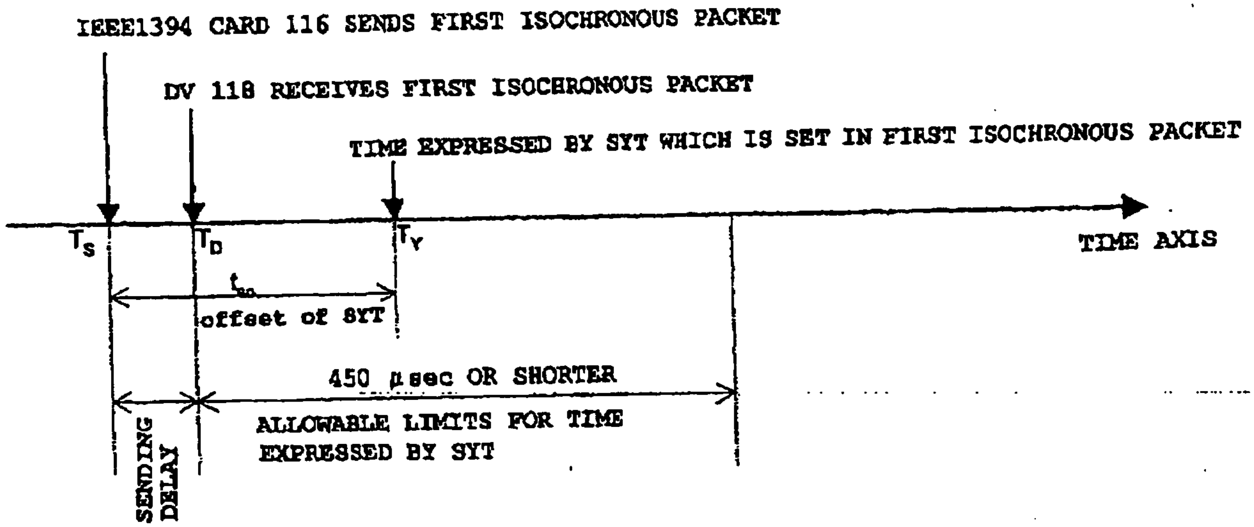

IEEE1394 CARD 116 SENDS FIRST ISOCHRONOUS PACKET
DV 118 RECEIVES FIRST ISOCHRONOUS PACKET
TIME EXPRESSED BY SYT WHICH IS SET IN FIRST ISOCHRONOUS PACKET
TIME AXIS
T_S
T_D
T_Y
t_{so}
offset of SYT
450 µsec OR SHORTER
SENDING DELAY
ALLOWABLE LIMITS FOR TIME EXPRESSED BY SYT

Fig. 11

FRAME NUMBER	FORMULA FOR CALCULATING SYT	SYT VALUE WHEN $Ts = 0\ \mu sec$ AND $tso = 250\ \mu sec$
0	$Ts+t_{so}$	2000_H
1	$T_S+t_{so}+1*1.001/30$	$CB34_H$
2	$T_S+t_{so}+2*1.001/30$	$7A6B_H$
3	$T_S+t_{so}+3*1.001/30$	$299C_H$
:	:	:
n	$T_S+t_{so}+n*1.001/30$	:
:	:	:

Fig. 12

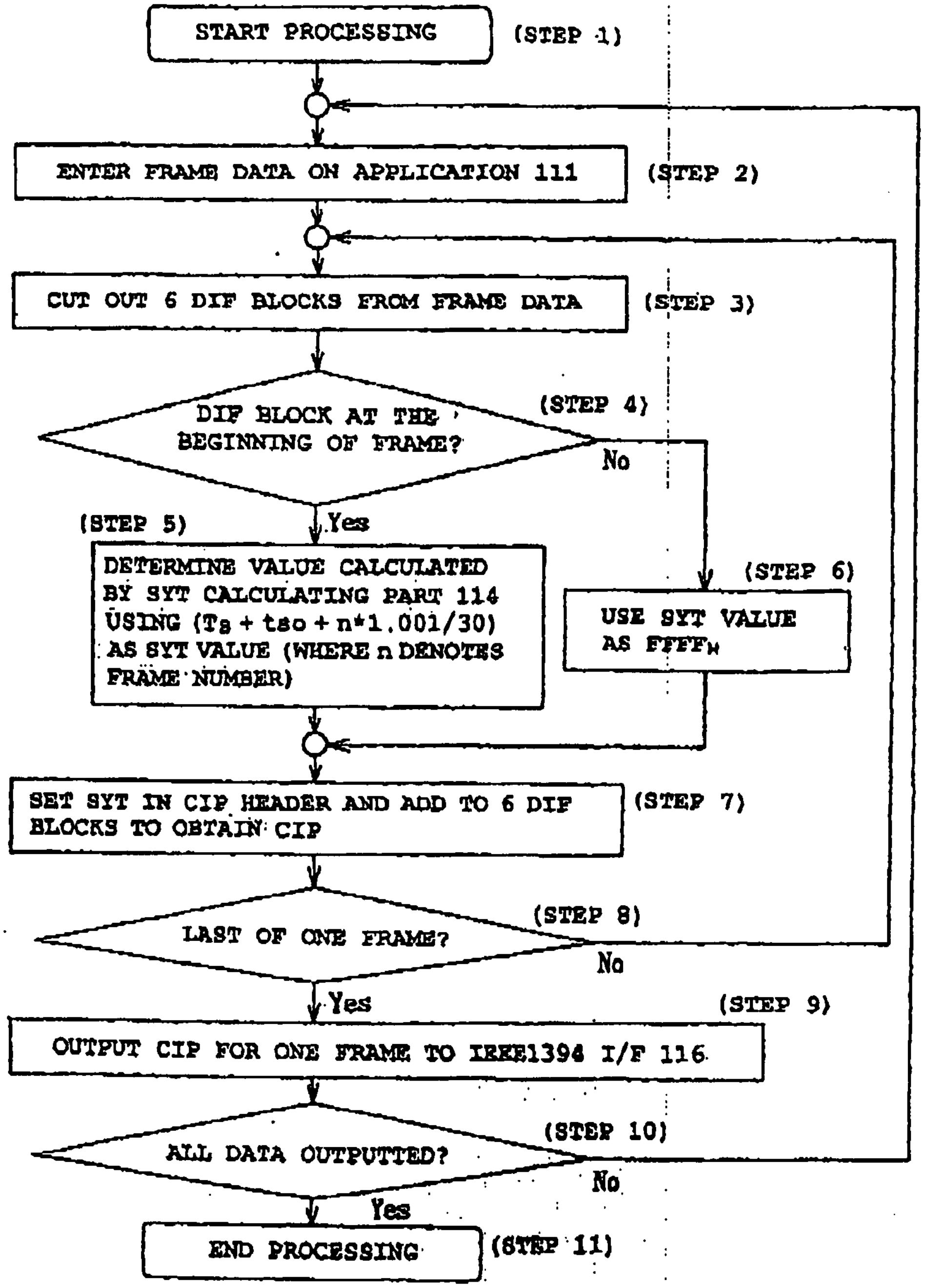

START PROCESSING
(STEP 1)
ENTER FRAME DATA ON APPLICATION 111
(STEP 2)
CUT OUT 6 DIF BLOCKS FROM FRAME DATA
(STEP 3)
DIF BLOCK AT THE BEGINNING OF FRAME?
(STEP 4)
No
Yes
(STEP 5)
DETERMINE VALUE CALCULATED BY SYT CALCULATING PART 114 USING (Ts + tso + n*1.001/30) AS SYT VALUE (WHERE n DENOTES FRAME NUMBER)
(STEP 6)
USE SYT VALUE AS FFFF$_H$
SET SYT IN CIP HEADER AND ADD TO 6 DIF BLOCKS TO OBTAIN CIP
(STEP 7)
LAST OF ONE FRAME?
(STEP 8)
No
Yes
(STEP 9)
OUTPUT CIP FOR ONE FRAME TO IEEE1394 I/F 116
ALL DATA OUTPUTTED?
(STEP 10)
No
Yes
END PROCESSING
(STEP 11)

Fig. 13

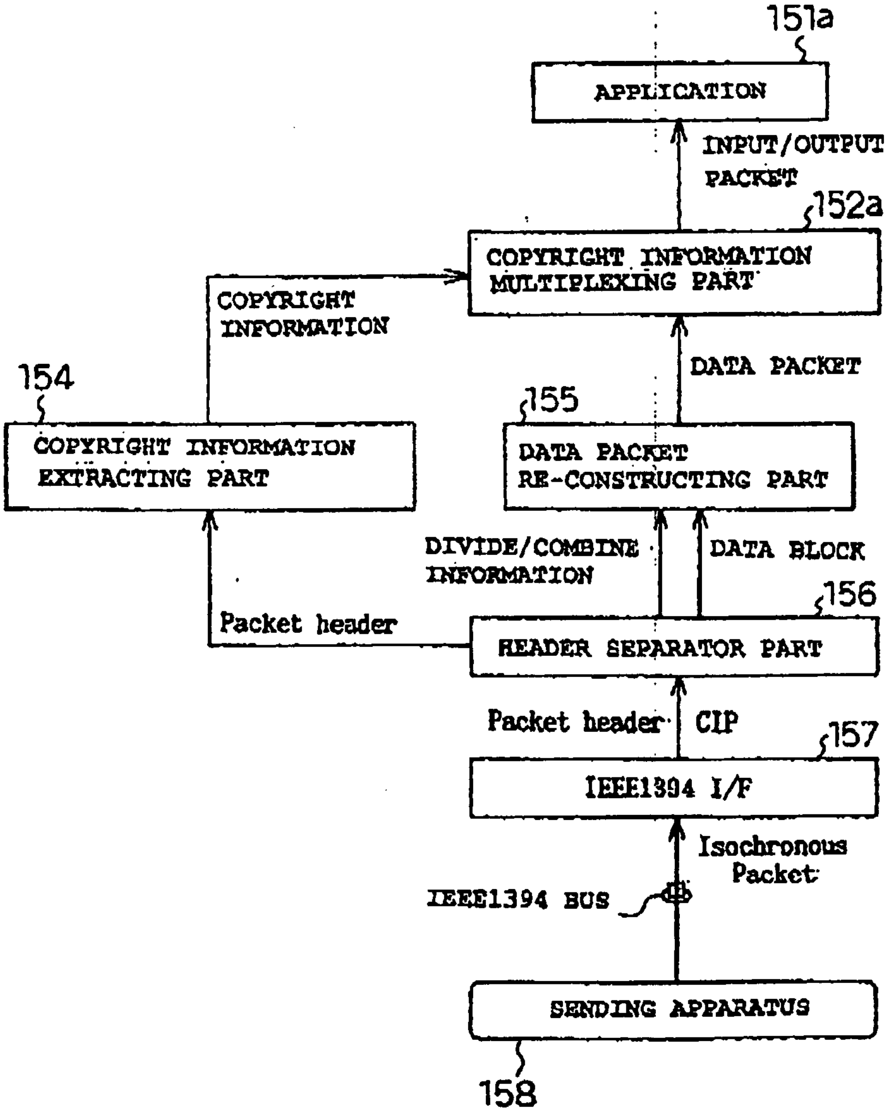

151a
APPLICATION
INPUT/OUTPUT PACKET
152a
COPYRIGHT INFORMATION MULTIPLEXING PART
COPYRIGHT INFORMATION
DATA PACKET
154
COPYRIGHT INFORMATION EXTRACTING PART
155
DATA PACKET RE-CONSTRUCTING PART
DIVIDE/COMBINE INFORMATION
DATA BLOCK
156
Packet header
HEADER SEPARATOR PART
Packet header
CIP
157
IEEE1394 I/F
Isochronous Packet
IEEE1394 BUS
SENDING APPARATUS
158

Fig. 14

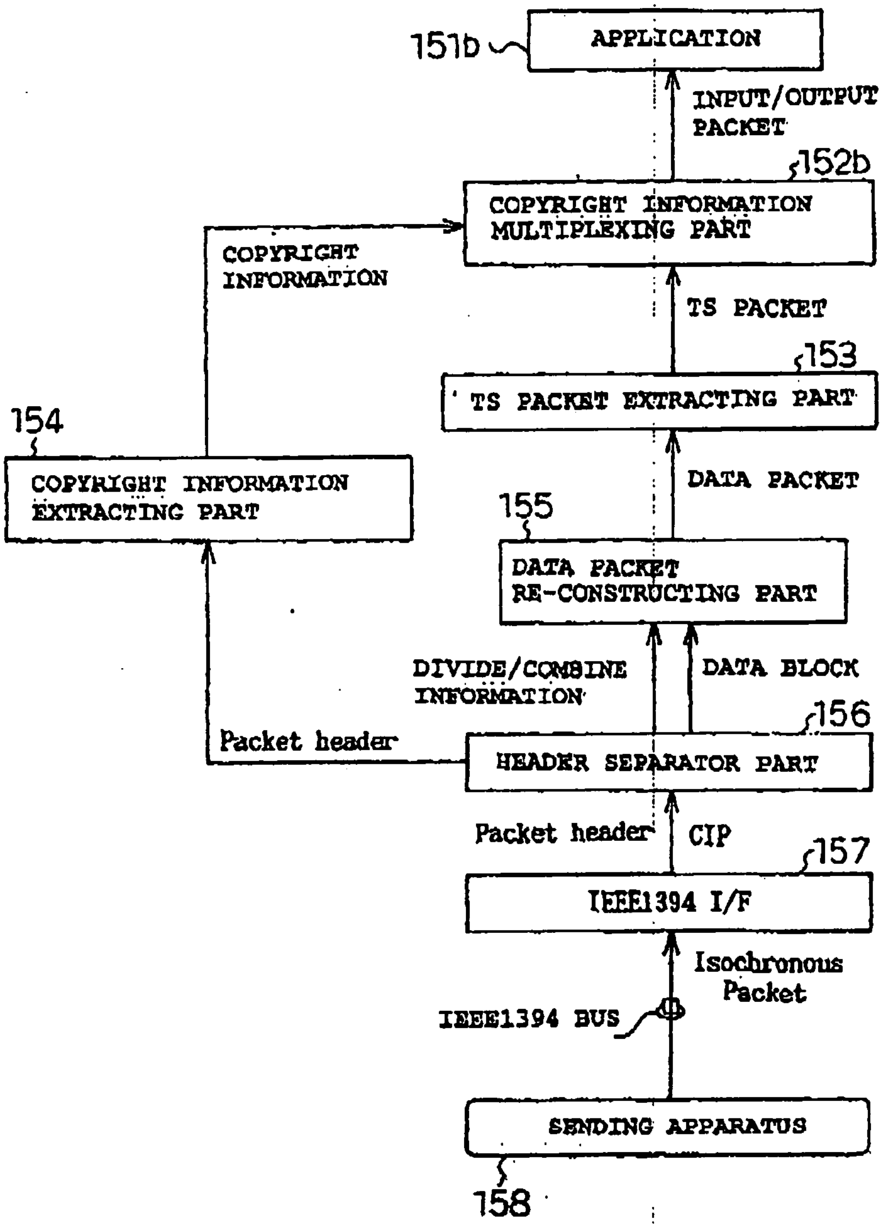

151b
APPLICATION
INPUT/OUTPUT PACKET
152b
COPYRIGHT INFORMATION MULTIPLEXING PART
COPYRIGHT INFORMATION
TS PACKET
153
TS PACKET EXTRACTING PART
154
COPYRIGHT INFORMATION EXTRACTING PART
DATA PACKET
155
DATA PACKET RE-CONSTRUCTING PART
DIVIDE/COMBINE INFORMATION
DATA BLOCK
156
Packet header
HEADER SEPARATOR PART
Packet header
CIP
157
IEEE1394 I/F
Isochronous Packet
IEEE1394 BUS
SENDING APPARATUS
158

Fig. 15

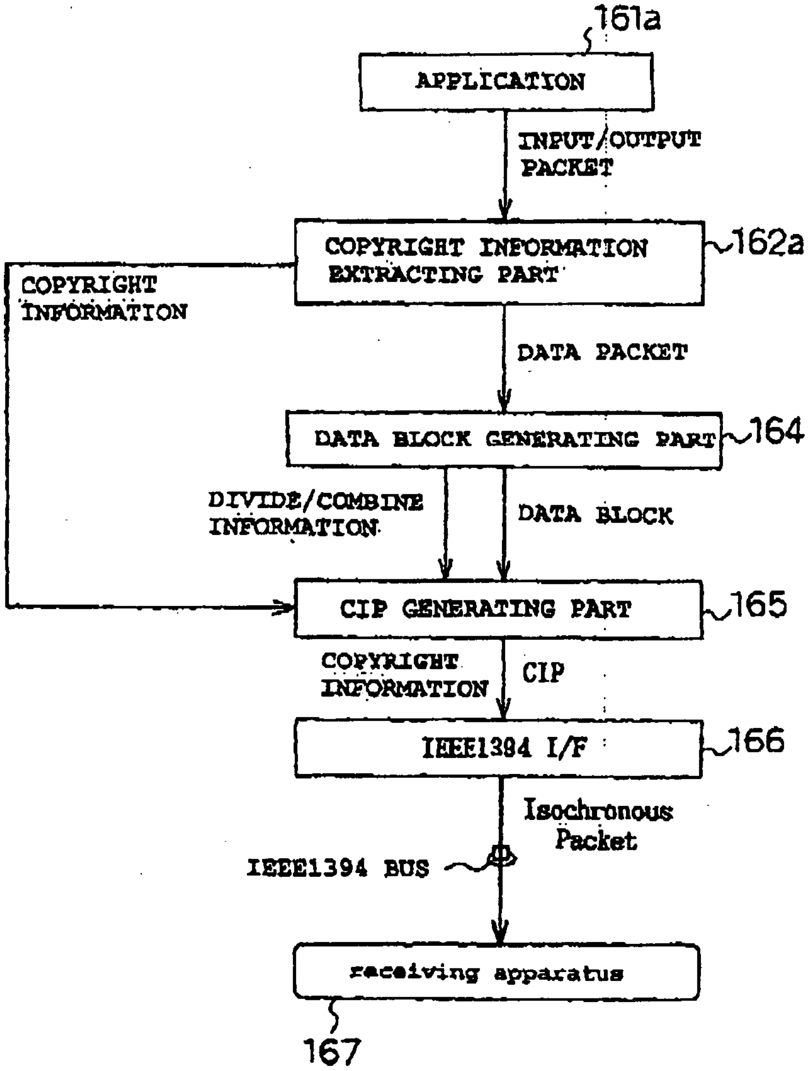
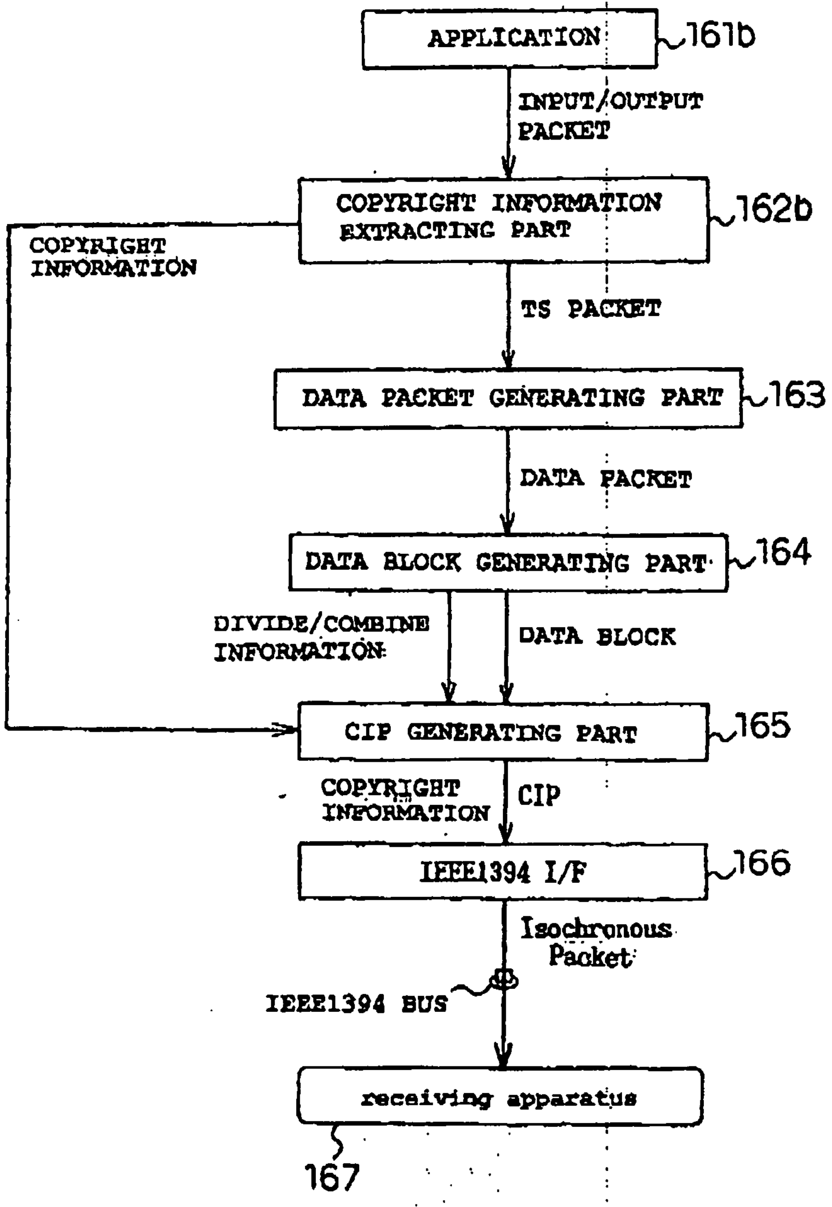

161a
APPLICATION
INPUT/OUTPUT PACKET
COPYRIGHT INFORMATION EXTRACTING PART
162a
COPYRIGHT INFORMATION
DATA PACKET
DATA BLOCK GENERATING PART
164
DIVIDE/COMBINE INFORMATION
DATA BLOCK
CIP GENERATING PART
165
COPYRIGHT INFORMATION
CIP
IEEE1394 I/F
166
Isochronous Packet
IEEE1394 BUS
receiving apparatus
167

Fig. 16


APPLICATION
161b
INPUT/OUTPUT PACKET
COPYRIGHT INFORMATION EXTRACTING PART
162b
COPYRIGHT INFORMATION
TS PACKET
DATA PACKET GENERATING PART
163
DATA PACKET
DATA BLOCK GENERATING PART
164
DIVIDE/COMBINE INFORMATION
DATA BLOCK
CIP GENERATING PART
165
COPYRIGHT INFORMATION
CIP
IEEE1394 I/F
166
Isochronous Packet
IEEE1394 BUS
receiving apparatus
167

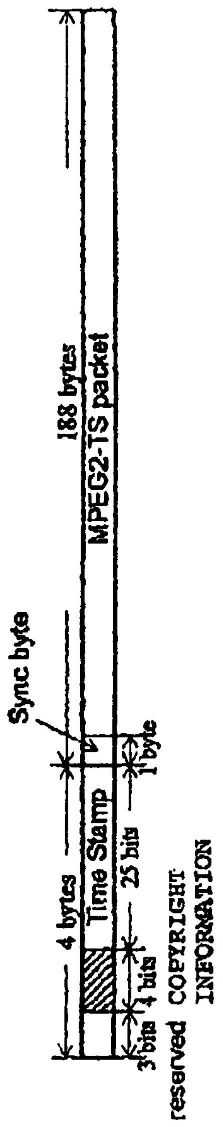

188 bytes
MPEG2-TS packet
Sync byte
1 byte
4 bytes
Time Stamp
25 bits
4 bits
3 bits
reserved
COPYRIGHT INFORMATION

Fig. 17 (a)

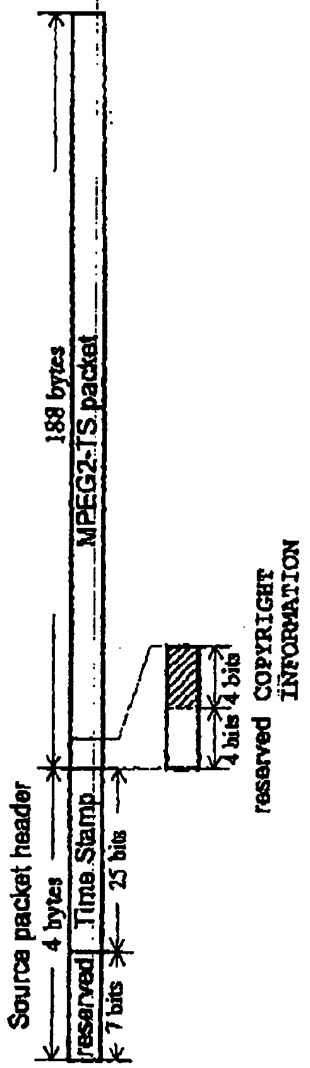

188 bytes
MPEG2-TS packet
Source packet header
4 bytes
Time Stamp
25 bits
reserved
7 bits
4 bits
4 bits
reserved
COPYRIGHT INFORMATION

Fig. 17 (b)

Fig. 17 (c)

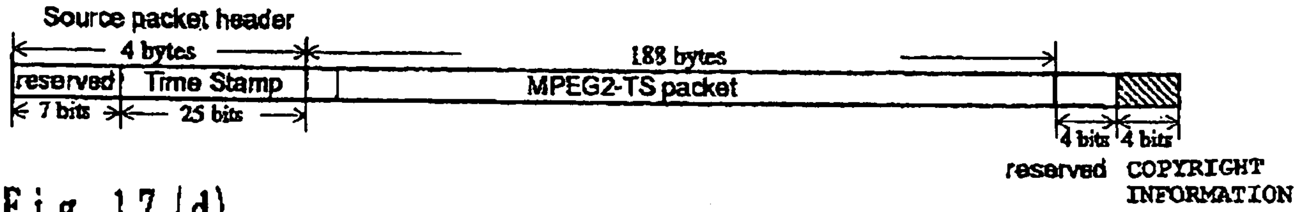

Source packet header
4 bytes
reserved
Time Stamp
7 bits
25 bits
188 bytes
MPEG2-TS packet
4 bits
4 bits
reserved
COPYRIGHT INFORMATION

Fig. 17 (d)

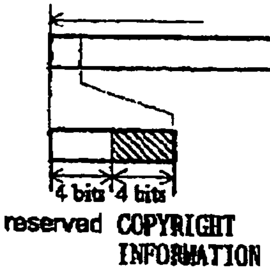
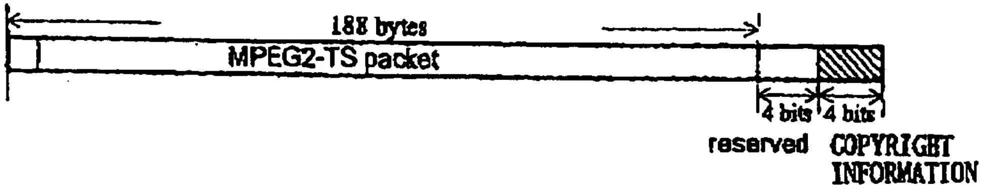

188 bytes
MPEG2-TS packet
4 bits
4 bits
reserved
COPYRIGHT INFORMATION

Fig. 17 (e)


188 bytes
MPEG2-TS packet
4 bits
4 bits
reserved
COPYRIGHT INFORMATION

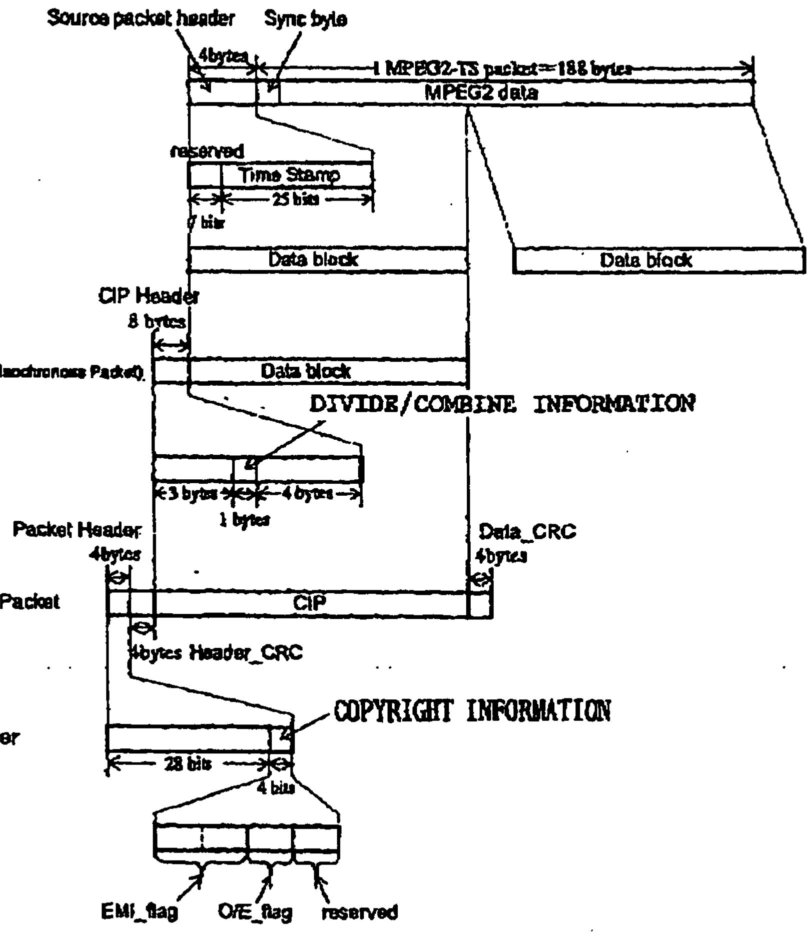

Source packet header
Sync byte
4bytes
1 MPEG2-TS packet=188 bytes
MPEG2 data
reserved
Time Stamp
7 bits
25 bits
Data block
Data block
CIP Header
8 bytes
Data block
DIVIDE/COMBINE INFORMATION
3 bytes
1 bytes
4 bytes
Packet Header
4bytes
Data_CRC
4bytes
CIP
4bytes Header_CRC
COPYRIGHT INFORMATION
28 bits
4 bits
EMI_flag
O/E_flag
reserved

Fig. 18 (a) Data packet

Fig. 18 (b) CIP (Common Isochronous Packet)

Fig. 18 (c) Isochronous Packet

Fig. 18 (d) Packet Header

Fig. 19 (a)

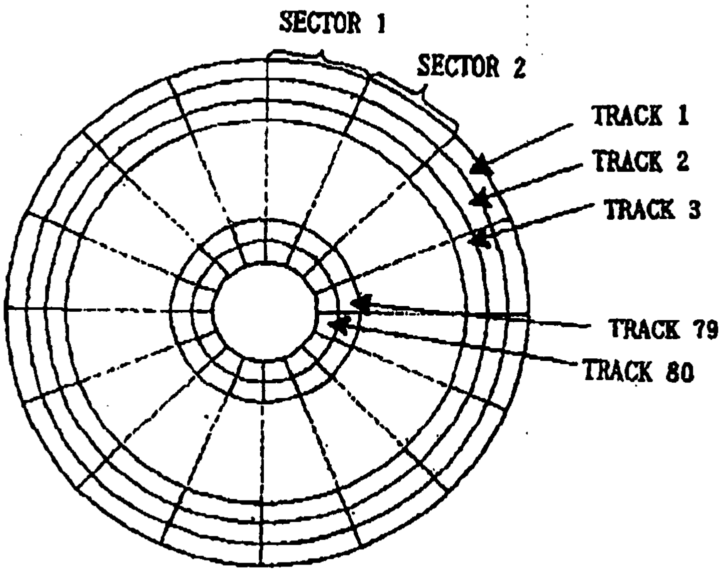

SECTOR 1
SECTOR 2
TRACK 1
TRACK 2
TRACK 3
TRACK 79
TRACK 80

Fig. 19(b)

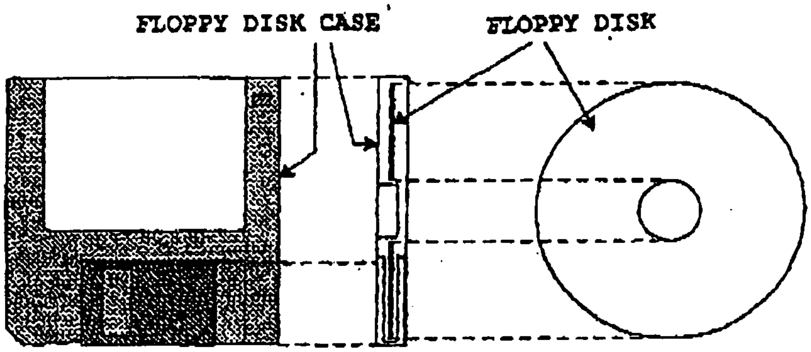
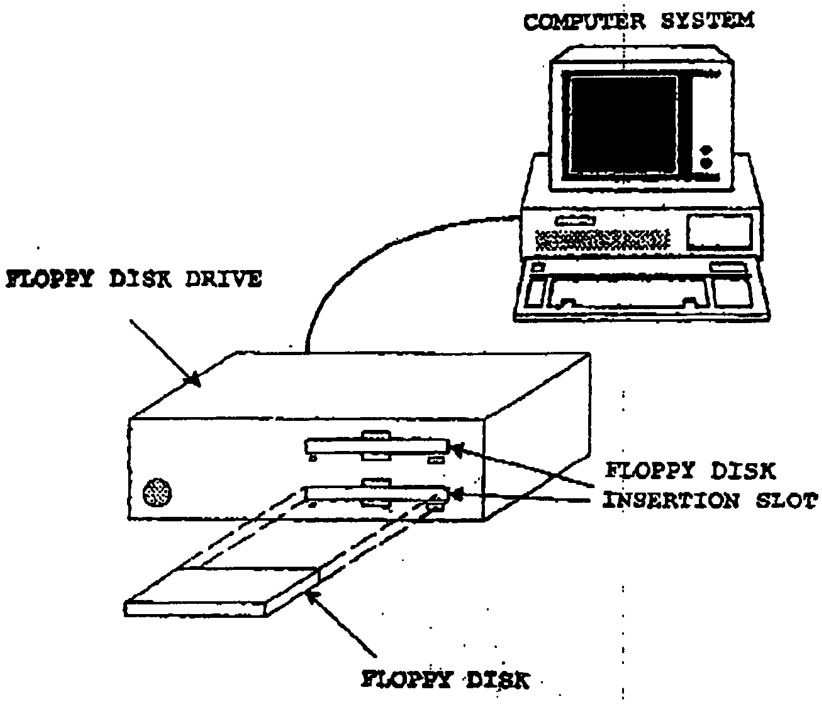

FLOPPY DISK CASE
FLOPPY DISK

Fig. 19 (c)


COMPUTER SYSTEM
FLOPPY DISK DRIVE
FLOPPY DISK INSERTION SLOT
FLOPPY DISK

INTERNATIONAL SEARCH REPORT

International application No.
PCT/JP00/01928

A. CLASSIFICATION OF SUBJECT MATTER
Int.Cl^7 G06F3/06

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
Int.Cl^7 G06F3/06
Int.Cl^7 H04N5/92

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Jitsuyo Shinan Koho	1926-1996	Jitsuyo Shinan Toroku Koho	1996-2000
Kokai Jitsuyo Shinan Koho	1971-2000	Toroku Jitsuyo Shinan Koho	1994-2000

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	JP, 11-53844, (Matsushita Electric Ind. Co., Ltd.), 26 February, 1999 (26.02.99) (Family: none) Full text; Figs. 1 to 8	1-12
X A	JP, 11-65773, (Matsushita Electric Ind. Co., Ltd.), 09 March, 1999 (09.03.99) (Family: none) Full text; Figs. 1 to 12	13-16,18-21, 24-26 17,22,23
X	JP, 10-302391, (Matsushita Electric Ind. Co., Ltd.), 13 November, 1998 (13.11.98) (Family: none) Full text; Figs. 1 to 2	27-49

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A" document defining the general state of the art which is not considered to be of particular relevance	"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E" earlier document but published on or after the international filing date	"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&" document member of the same patent family
"O" document referring to an oral disclosure, use, exhibition or other means	
"P" document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search 28 June, 2000 (28.06.00)	Date of mailing of the international search report 18 July, 2000 (18.07.00)
Name and mailing address of the ISA/ Japanese Patent Office	Authorized officer
Facsimile No.	Telephone No.

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

International application No.
PCT/JP00/01928

Box I Observations where certain claims were found unsearchable (Continuation of item 1 of first sheet)

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claims Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claims Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. ☐ Claims Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box II Observations where unity of invention is lacking (Continuation of item 2 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

The inventions of claims 1 to 12 relate to a device driver for handling real time data.
The inventions of claims 13 to 49 relate to a packet transmitted/received through an IEEE-1394 interface.

1. ☐ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.

2. ☒ As all searchable claims could be searched without effort justifying an additional fee, this Authority did not invite payment of any additional fee.

3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claims Nos.:

4. ☐ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claims Nos.:

Remark on Protest ☐ The additional search fees were accompanied by the applicant's protest.
☐ No protest accompanied the payment of additional search fees.

Form PCT/ISA/210 (continuation of first sheet (1)) (July 1992)

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

☐ **BLACK BORDERS**

☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**

☐ **FADED TEXT OR DRAWING**

☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**

☐ **SKEWED/SLANTED IMAGES**

☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**

☐ **GRAY SCALE DOCUMENTS**

☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**

☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**

☐ **OTHER:** ______________________

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.